R01AN2155EJ0100
Rev.1.00
9. 26, 2014

RX62T

Vector control of permanent magnetic synchronous motor (Implementation)

Abstract

This application note aims at explaining sample programs for operating vector control of a permanent magnetic synchronous motor, by using functions of RX62T, and how to use a library of the development support tool, In Circuit Scope.

The sample programs should be used just as reference and Renesas Electronics does not guarantee the operations. Please use these sample programs after carrying out a thorough evaluation in a suitable environment.

In particular, handling the high voltage environment is extremely dangerous. Before using each development environment, read the user's manuals carefully. Renesas Electronics assumes no liability whatsoever for any damages arising from the use of development environment described in this application note.

Operation checking device

Operations of the sample programs are checked by using the following device.

• RX62T (R5F562TAADFP)

Target sample programs

The target sample programs of this application note are as follows.

(1) RX62T100_T2001_SPM_LESS_FOC_CSP_V100

Sensorless vector control sample program for RX62T (R5F562TAADFP), T2001

(2) RX62T100_T1102_SPM_LESS_FOC_CSP_V100

Sensorless vector control sample program for RX62T (R5F562TAADFP), T1102

Reference documents

• RX62T Group User's Manual: Hardware (R01UH0034EJ0200)

• Application note: 'Vector control of permanent magnetic synchronous motor: algorithm'.

• In Circuit Scope Manual

Downloadable from: http://www.desktoplab.co.jp/download.html

• Trial series "T1102" 3kW 4kVA Inverter Unit User's Manual

• Trial series "T2001" 50W 60VA Low Voltage Inverter Unit User's Manual

• Trial series "T5201" RX62T 100pin CPU card User's Manual

Contents

1. Overview

This application note explains how to implement the vector control sample programs of permanent magnetic synchronous motor (PMSM) using the RX62T microcontroller and how to use the library of the development support tool, In Circuit Scope (hereinafter referred to as ICS) (Note 1). Note that these sample programs use the algorithm described in the application note 'Vector control of permanent magnetic synchronous motor: algorithm'

1.1 Development environment

Table 1-1 shows development environment of the sample programs explained in this application note.

Table 1-1 Development Environment of the Sample Programs

	Sample program	Microcontroller	Inverter board	Motor	Version of CubeSuite+
Low-voltage version	(1)	R5F562TAADFP	T2001 (Note 1)	FH6S20E-X81 (Note 2)	V2.02.00
High-voltage version	(2)	R5F562TAADFP	T1102 (Note 1)	BXM6200-A (Note 3)	V2.02.00

For purchase and technical support of inverter boards T2001/T1102, contact sales representatives and dealers of Renesas Electronics Corporation.

Notes:

1. The inverter board T2001, T1102, and the development support tool In Circuit Scope are the products of Desk Top Laboratories Inc.

   Desk Top Laboratories Inc. (http://www.desktoplab.co.jp/)

2. FH6S20E-X81 is the product of NIDEC SERVO CORPORATION.

   NIDEC SERVO CORPORATION. (http://www.nidec-servo.com/en/index.html)

3. BXM6200-A is the product of ORIENTAL MOTOR CO., LTD.

   ORIENTAL MOTOR CO., LTD. (http://www.orientalmotor.co.jp/)

2. System overview

Overview of this system is explained below.

2.1 Hardware configuration

The hardware configuration is shown below.

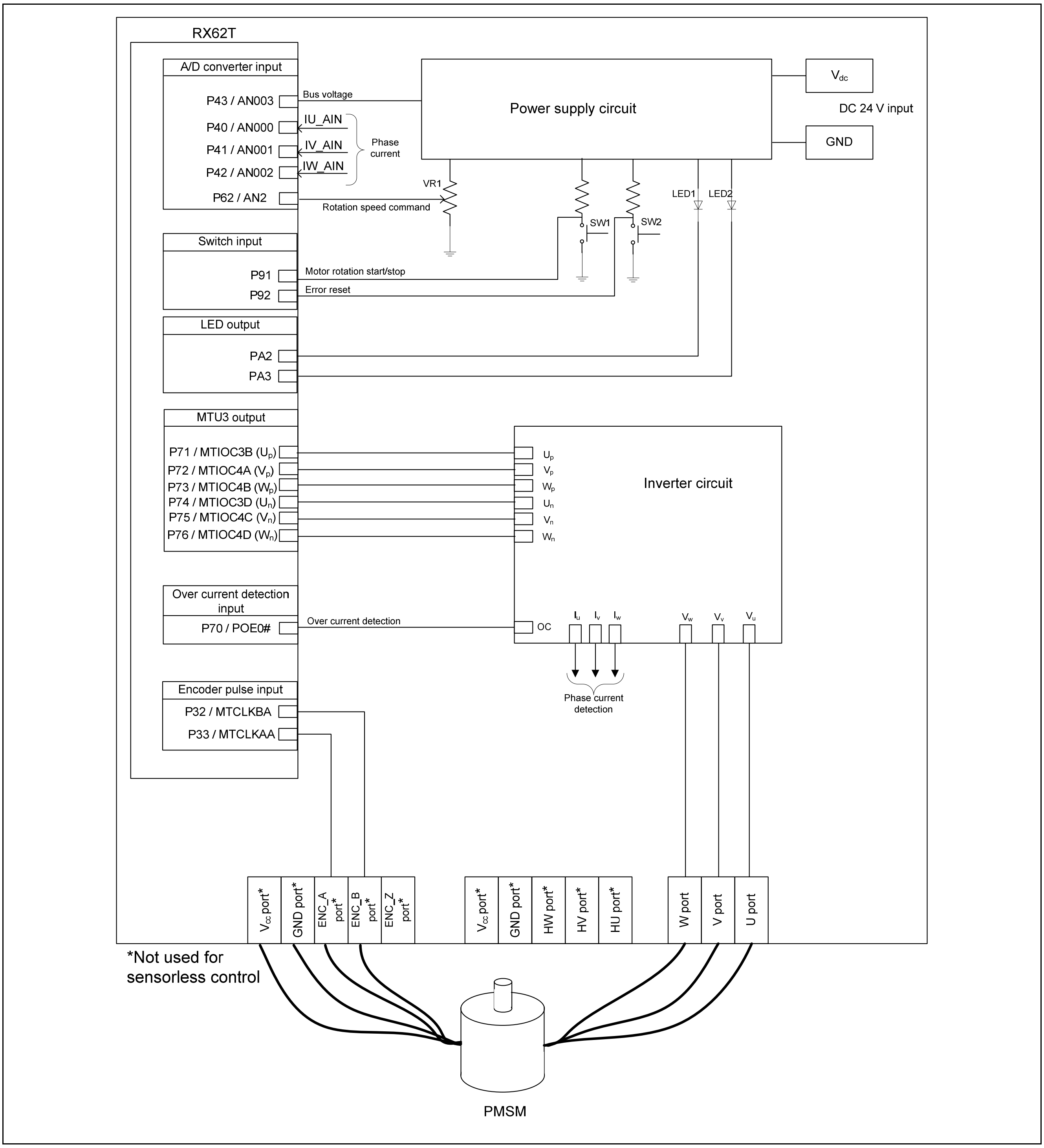

Figure 2-1 Hardware Configuration Diagram (R5F562TAADFP + T2001/Target Software: (1))

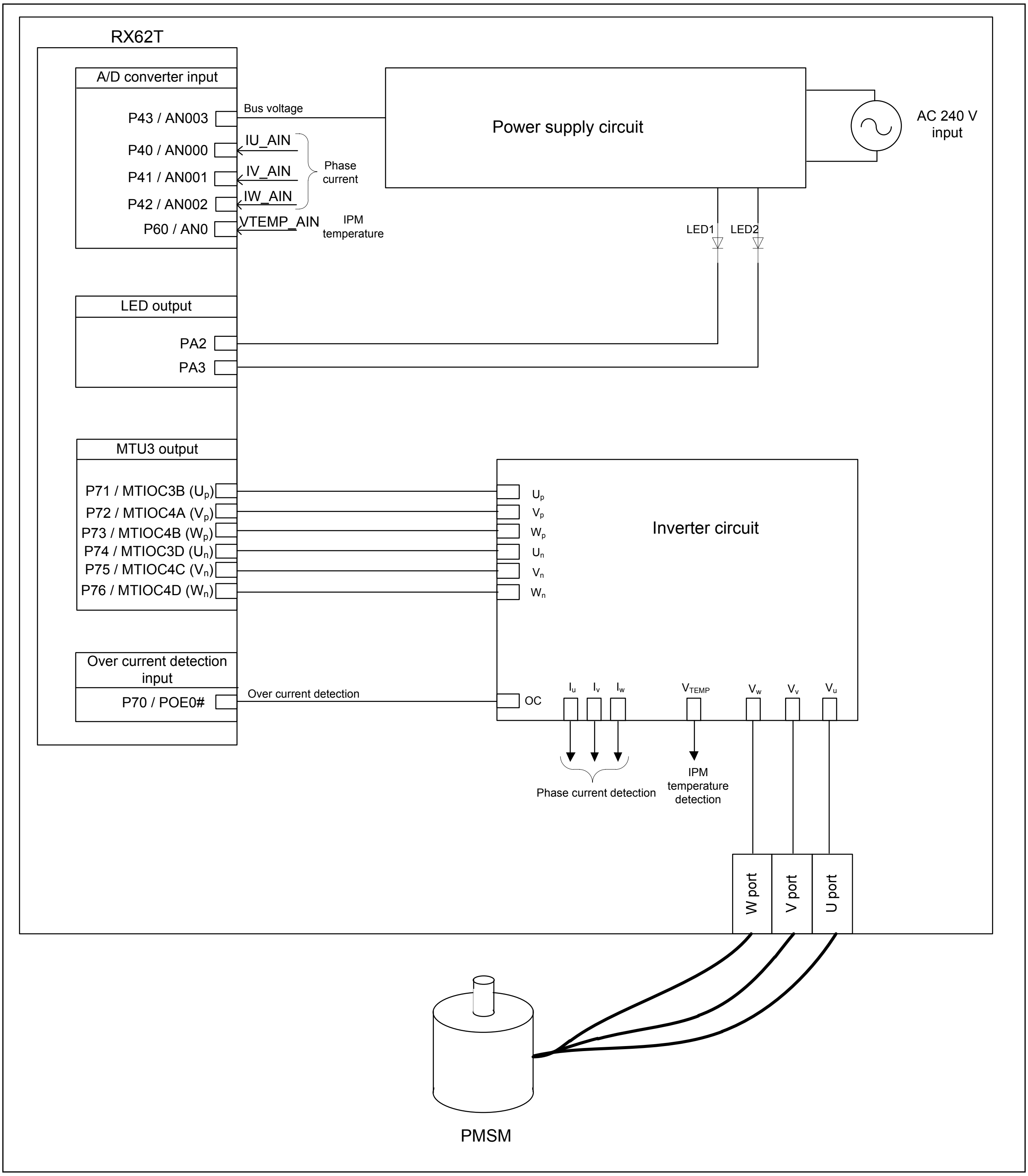

Figure 2-2 Hardware Configuration Diagram (R5F562TAADFP + T1102/Target Software: (2))

2.2 Hardware specifications

2.2.1 User interface

List of user interfaces of this system is given in Table 2-1.

Table 2-1 User Interface

Item	Interface component		Function
	Target software (1)	Target software (2)	
Rotation speed	Variable resistor (VR1)	—	Reference value of rotation speed input (analog value)
START/STOP	Toggle switch (SW1)	—	Motor rotation start/stop command
ERROR RESET	Toggle switch (SW2)	—	Command of recovery from error status
LED1	Yellow green LED	Yellow green LED	・At the time of motor rotation: ON ・At the time of stop: OFF
LED2	Yellow green LED	Yellow green LED	・At the time of error detection: ON ・At the time of normal operation: OFF
RESET	Push switch (RESET1)	Push switch (RESET1)	System reset

List of port interfaces of this system is given in Table 2-2.

Table 2-2 Port Interfaces

R5F562TAADFP port name		Function
Target software (1)	Target software (2)	
P43 / AN003	P43 / AN003	Inverter bus voltage measurement
P62 / AN2	-	For rotation speed command value input (analog value)
P91	-	START/STOP toggle switch
P92	-	ERROR RESET toggle switch
PA2	PA2	LED1 ON/OFF control
PA3	PA3	LED2 ON/OFF control
P40 / AN000	P40 / AN000	U phase current measurement
P41 / AN001	P41 / AN001	V phase current measurement
P42 / AN002	P42 / AN002	W phase current measurement
-	P60 / AN0	IPM temperature measurement
P71 / MTIOC3B	P71 / MTIOC3B	Complementary PWM output (U_p)
P72 / MTIOC4A	P72 / MTIOC4A	Complementary PWM output (V_p)
P73 / MTIOC4B	P73 / MTIOC4B	Complementary PWM output (W_p)
P74 / MTIOC3D	P74 / MTIOC3D	Complementary PWM output (U_n)
P75 / MTIOC4C	P75 / MTIOC4C	Complementary PWM output (V_n)
P76 / MTIOC4D	P76 / MTIOC4D	Complementary PWM output (W_n)
P33 / MTCLKAA	-	Encoder A phase input
P32 / MTCLKBA	-	Encoder B phase input
P70 / POE0#	P70 / POE0#	PWM emergency stop input at the time of overcurrent detection

2.2.2 Peripheral functions

List of the peripheral functions used in this system is given in Table 2-3.

Table 2-3 List of the Peripheral Functions for Each Sample Program

	12-bit A/D	10-bit A/D	CMT	MTU3	POE3
(1)	• Current of each phase U, V, and W • Inverter bus voltage	Rotation speed command value	1 [ms] interval timer	Complementary PWM output	Initialization of complementary PWM output ports (Set PWM output ports to high impedance state to stop the PWM output.)
(2)		IPM temperature			

1. 12-bit A/D converter

U phase current (Iu), V phase current (Iv), W phase current (Iw), and inverter bus voltage (Vdc) are measured by using the 12-bit A/D converter.

The operation mode is set the single-cycle scan mode with the sample-and- hold function (use hardware trigger).

2. 10-bit A/D converter

The rotation speed command value and IPM temperature are measured by using the 10-bit A/D converter. Set the operation mode to the single mode. (use software trigger).

3. Compare match timer (CMT)

The channel 0 of the compare match timer is used as 1 [ms] interval timer.

4. Multi-function timer pulse unit 3 (MTU3)

The operation mode varies depending on channels. On the channels 3 and 4, output with dead time (high active) is performed by using the complementary PWM mode.

5. Port output enable 3 (POE3)

The PWM output ports are set to high impedance state to stop the PWM output and initialize the complementary PWM output port when the over current is detected (when a falling edge of the POE0# port is detected) and when the output short circuit is detected.

2.3 Software configuration

2.3.1 Software file configuration

Folder and file configuration of the sample programs are given below.

Table 2-4 Folder and File Configuration of the Sample Program (Target Software: (1))

RX62T100_T2001_SPM_LESS _FOC_CSP_V100	inc	main.h	Main function, user interface control header
		mtr_common.h	Common definition header
		mtr_ctrl_t2001.h	Board dependent processing part header
		mtr_ctrl_rx62t100.h	RX62T dependent processing part header
		mtr_spm_less_foc.h	Sensorless vector control header
		control_parameter.h	Header for control parameter
		motor_parameter.h	Header for motor parameter
		mtr_ctrl_rx62t100_t2001.h	Board and RX62T dependent processing part header
	lib	angle_speed_R5F562TAAxFP.obj	Angle and speed estimation library
	ics	ICS_RX62T.obj	ICS library
		ICS_RX62T.h	Header for ICS
	src	main.c	Main function, user interface control
		mtr_ctrl_t2001.c	Board dependent processing
		mtr_ctrl_rx62t100.c	RX62T dependent processing
		mtr_interrupt.c	Interrupt handler
		mtr_spm_less_foc.c	Sensorless vector control
		mtr_ctrl_rx62t100_t2001.c	Board and RX62T dependent processing

Table 2-5 Folder and File Configuration of the Sample Program (Target Software: (2))

RX62T100_T1102_SPM_LESS _FOC_CSP_V100	inc	main.h	Main function, user interface control header
		mtr_common.h	Common definition header
		mtr_ctrl_t1102.h	Board dependent processing part header
		mtr_ctrl_rx62t100.h	RX62T dependent processing part header
		mtr_spm_less_foc.h	Sensorless vector control header
		control_parameter.h	Header for control parameter
		motor_parameter.h	Header for motor parameter
		mtr_ctrl_rx62t100_t1102.h	Board and RX62T dependent processing part header
	lib	angle_speed_R5F562TAAxFP.obj	Angle and speed estimation library
	ics	ICS_RX62T.obj	ICS library
		ICS_RX62T.h	Header for ICS
	src	main.c	Main function, user interface control
		mtr_ctrl_t1102.c	Board dependent processing
		mtr_ctrl_rx62t100.c	RX62T dependent processing
		mtr_interrupt.c	Interrupt handler
		mtr_spm_less_foc.c	Sensorless vector control
		mtr_ctrl_rx62t100_t1102.c	Board and RX62T dependent processing

2.3.2 Module configuration

Module configuration of the sample programs is described below.

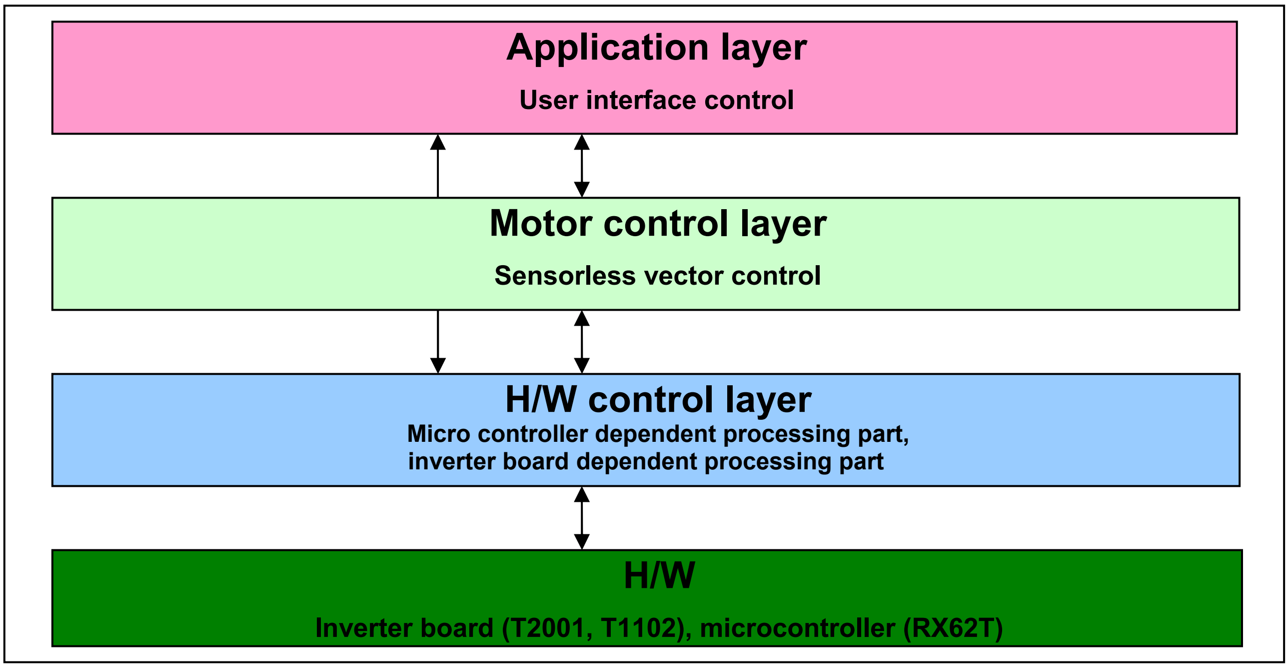

Figure 2-3 Module Configuration of the Sample Programs

Table 2-6 Module Structure of the Sample Programs

	(1)	(2)
Application layer	main.c	main.c
Motor control layer	mtr_spm_less_foc.c	mtr_spm_less_foc.c
H/W control layer	mtr_ctrl_ rx62t100_t2001.c mtr_ctrl_ rx62t100.c mtr_ctrl_t2001.c	mtr_ctrl_ rx62t100_t1102.c mtr_ctrl_ rx62t100.c mtr_ctrl_t1102.c

2.4 Software specifications

Table 2-7 shows basic software specification of this system. For details of the vector control, refer to the application note ‘Vector control of permanent magnetic synchronous motor: algorithm’.

Table 2-7 Basic Specifications of Sensorless Vector Control Software (Target Software: (1))

Item	Content
Control method	Vector control
Motor rotation start/stop	Determined depending on the level of SW1 (P91 (”Low”: rotation start “High”: stop) or input from ICS (Note 1)
Position detection of rotor magnetic pole	Sensorless
Input voltage	DC 24 V
Carrier frequency (PWM)	20 [kHz]
Control cycle	100 [μs] (twice the carrier cycle)
Rotation speed control range	CW: 0 [rpm] to 2000 [rpm] CCW: 0 [rpm] to 2000 [rpm]
Processing stop for protection	・Disables the motor control signal output (six outputs), under any of the following four conditions. 1. Current of each phase exceeds 4.8 [A] (monitored every 100 [μs]) 2. Inverter bus voltage exceeds 28 [V] (monitored every 100 [μs]) 3. Inverter bus voltage is less than 0 [V] (monitored every 100 [μs]) 4. Rotation speed exceeds 4000 [rpm] (monitored every 100 [μs]) ・When an external over current signal is detected (when a falling edge of the POE0# port is detected) and when the output short circuit is detected, the PWM output ports are set to high impedance state.

Table 2-8 Basic Specifications of Sensorless Vector Control Software (Target Software: (2))

Item	Content
Control method	Vector control
Motor rotation start/stop	Input from ICS (Note 1)
Position detection of rotor magnetic pole	Sensorless
Input voltage	AC 240 V (without PFC)
Carrier frequency (PWM)	20 [kHz]
Control cycle	100 [μs] (twice the carrier cycle)
Rotation speed control range	CW: 0 [rpm] to 3000 [rpm] CCW: 0 [rpm] to 3000 [rpm]
Processing stop for protection	・Disables the motor control signal output (six outputs), under any of the following four conditions. 1. Current of each phase exceeds 4.8 [A] (monitored every 100 [μs]) 2. Inverter bus voltage exceeds 400 [V] (monitored every 100 [μs]) 3. Inverter bus voltage is less than 100 [V] (monitored every 100 [μs]) 4. Rotation speed exceeds 6000 [rpm] (monitored every 100 [μs]) 5. IPM temperature output value exceeds 3 [V] (60 ± 10 [°C]) (monitored every 100 [μs]) ・When an external over current signal is detected (when a falling edge of the POE0# port is detected) and when the output short circuit is detected, the PWM output ports are set to high impedance state.

Note:

1. For more details, refer to 4. Development support tool In Circuit Scope.

3. Descriptions of the control program

The target sample programs of this application note are explained here.

3.1 Contents of control

3.1.1 Motor start/stop

Starting and stopping of the motor are controlled by input from SW1 or ICS.

A general-purpose port (P91) is assigned to SW1. The P91 port is read within the main loop. When P91 is at a "Low" level, it is determined that the start switch is being pressed. Conversely, when the level is switched to "High", the program determines that the motor should be stopped.

3.1.2 Motor rotation speed command value, inverter bus voltage, and motor three-phase current

(1) Motor rotation speed command value

The motor rotation speed command value can be set by A/D conversion of the VR1 output value (analog value) or input from ICS. The A/D converted VR1 value is used as rotation speed command value, as shown in Table 3-1.

Table 3-1 Conversion Ratio of the Rotation Speed Command Value

<table>
<tr><th>Item</th><th>Sample software</th><th colspan="2">Conversion ratio
(Command value: A/D conversion value)</th><th>Channel</th></tr>
<tr><td rowspan="3">Rotation speed command value</td><td rowspan="2">(1)</td><td>CW</td><td>0 [rpm] to 2000 [rpm]: 0200H to 03FFH</td><td rowspan="2">AN2</td></tr>
<tr><td>CCW</td><td>0 [rpm] to 2000 [rpm]: 01FFH to 0000H</td></tr>
<tr><td>(2)
(Not used)</td><td colspan="2">-</td><td>-</td></tr>
</table>

(2) Inverter bus voltage

Inverter bus voltage is measured as given in Table 3-2.

It is used for modulation factor calculation and over voltage detection. (When an abnormality is detected, PWM is stopped.)

Table 3-2 Inverter Bus Voltage Conversion Ratio

<table>
<tr><th>Item</th><th>Sample software</th><th>Conversion ratio
(Inverter bus voltage: A/D conversion value)</th><th>Channel</th></tr>
<tr><td rowspan="2">Inverter bus voltage</td><td>(1)</td><td>0 [V] to 111 [V]: 0000H to 0FFFH</td><td rowspan="2">AN003</td></tr>
<tr><td>(2)</td><td>0 [V] to 686.5 [V]: 0000H to 0FFFH</td></tr>
</table>

(3) U, W phase current

The U and W phase currents are measured as shown in Table 3-3 and used for vector control.

Table 3-3 Conversion Ratio of U and W Phase Current

<table>
<tr><th>Item</th><th>Sample software</th><th>Conversion ratio
(U, W phase current: A/D conversion value)</th><th>Channel</th></tr>
<tr><td rowspan="2">U, W phase current</td><td>(1)</td><td>-10 [A] to 10 [A]: 0000H to 0FFFH</td><td rowspan="2">Iu: AN000
Iw: AN002</td></tr>
<tr><td>(2)</td><td>-50 [A] to 50 [A]: 0000H to 0FFFH</td></tr>
</table>

(4) IPM temperature

The IPM temperature is measured as shown in Table 3-4 and used for IPM temperature error detection.

For the relation of IPM temperature and the voltage, refer to the datasheet of IPM.

Table 3-4 Conversion Ratio of IPM temperature

Item	Sample software	Conversion ratio (IPM temperature: A/D conversion value)	Channel
IPM temperature	(1)(Not used)	-	-
	(2)	0 [V] to 5 [V]: 0000H to 03FFH	AN0

3.1.3 Modulation

The target sample software of this application note uses pulse width modulation (hereinafter called PWM) and the triangular wave comparison method to generate the input voltage to the motor and the PWM waveform respectively.

(1) Triangular wave comparison method

In order to actually output the command value voltage, the triangular wave comparison method is used. By this method, the pulse width of the output voltage can be determined by comparing the carrier waveform (triangular wave) and voltage command value waveform. Output of the voltage command value of the pseudo sinusoidal wave can be performed by turning the switch on or off when the command value voltage is larger or smaller than the carrier wave voltage respectively.

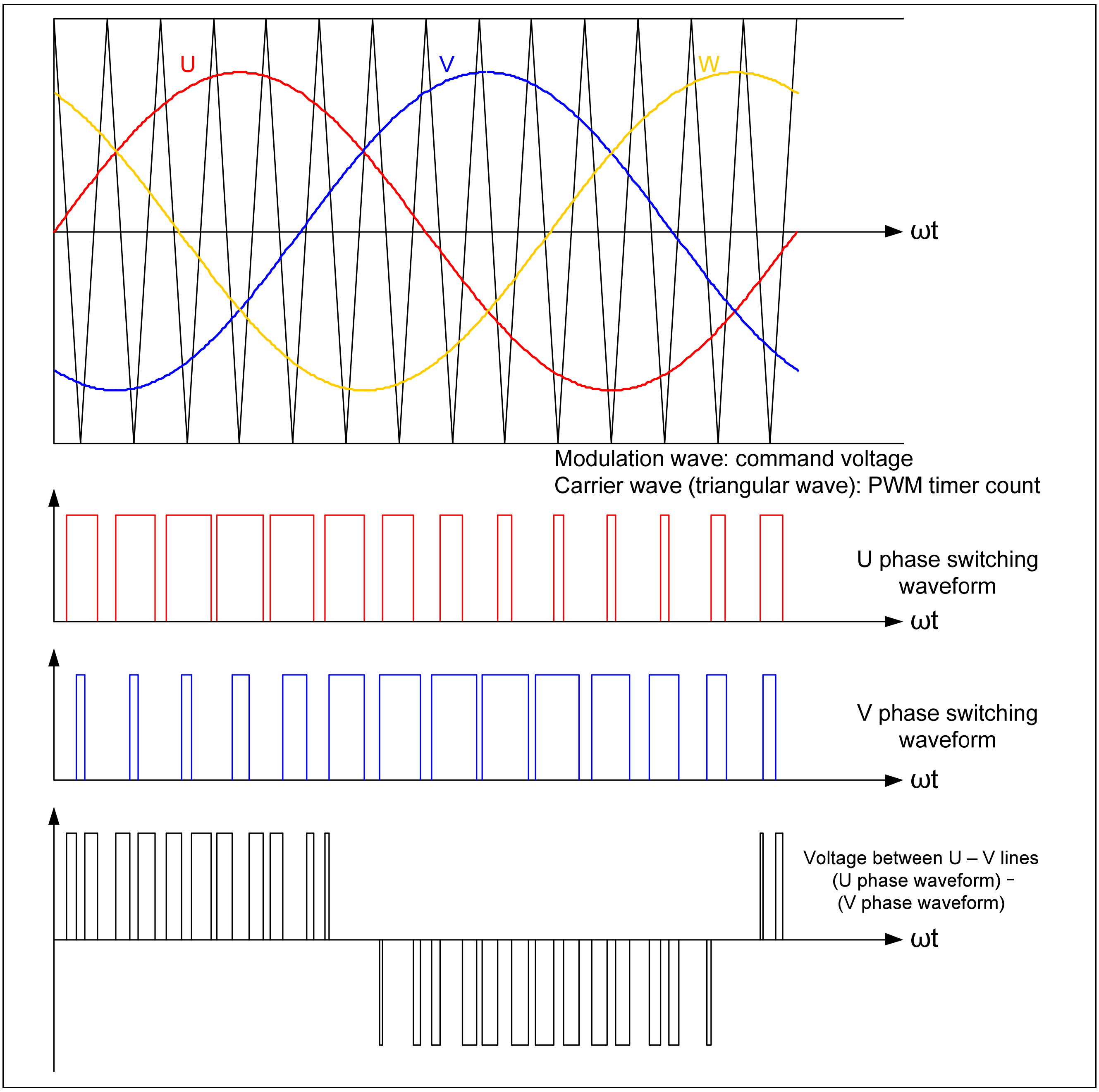

Figure 3-1 Conceptual Diagram of the Triangular Wave Comparison Method

Here, as shown in the Figure 3-2, ratio of the output voltage pulse to the carrier wave is called duty.

$$\text{Duty} = \frac{T_{ON}}{T_{ON} + T_{OFF}} \times 100\ [\%]$$

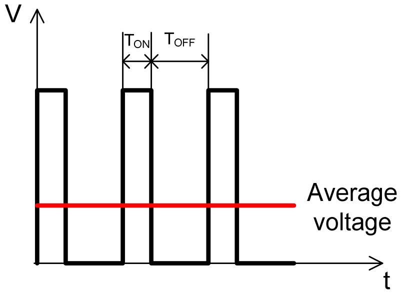

Figure 3-2 Definition of Duty

Modulation factor m is defined as follows.

$$m = \frac{V}{E}$$

m: Modulation factor V: Command value voltage E: Inverter bus voltage

A requested control can be performed by setting this modulation factor to the register which determines PWM duty.

3.1.4 State transition

Figure 3-3 is a state transition diagram of the sensorless vector control software.

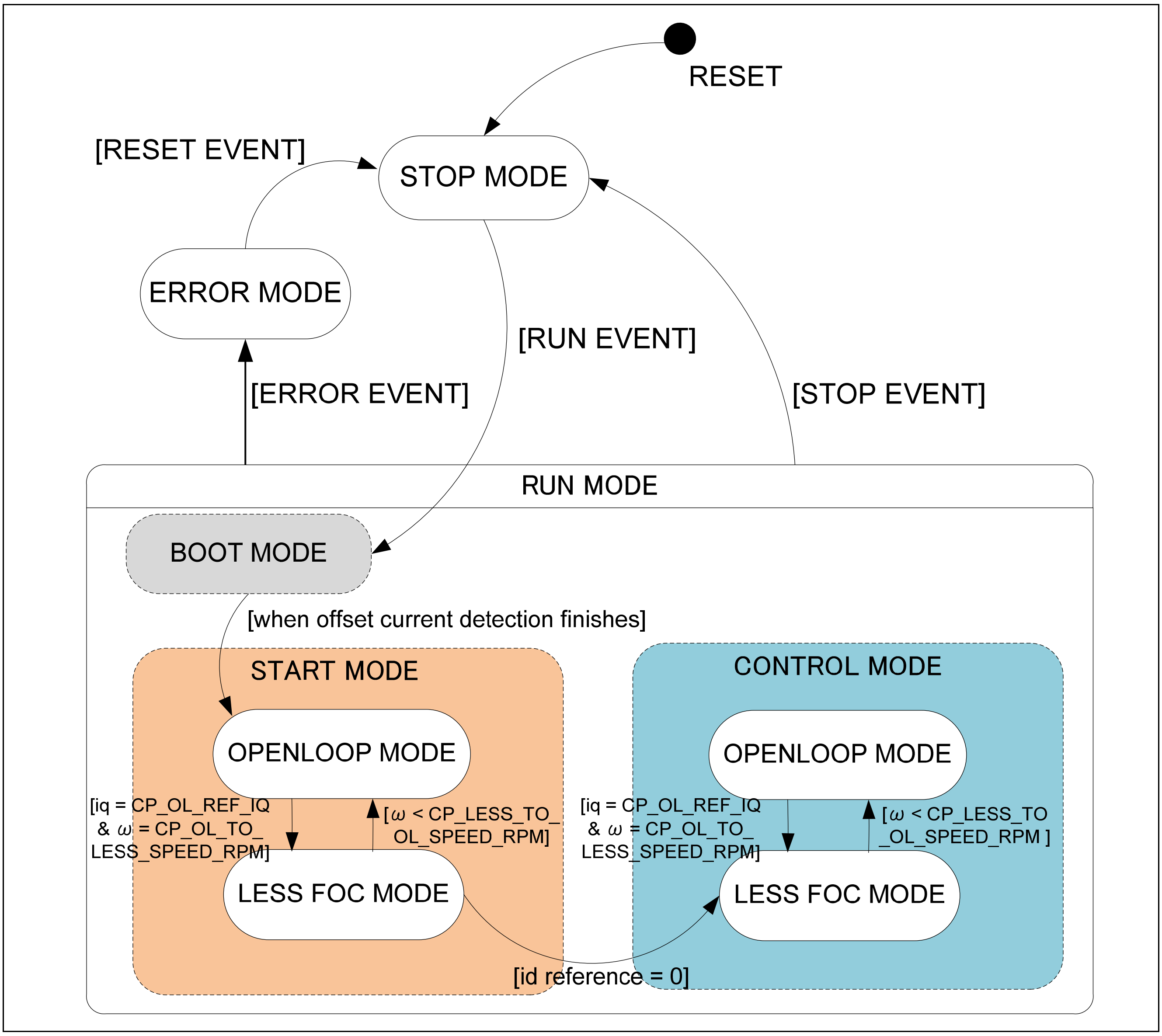

Figure 3-3 State Transition Diagram of Sensorless Vector Control Software (Target Software: (1), (2))

Figure 3-4 shows startup control of sensorless vector control software. Each mode is controlled by flags managing each command value of the d axis current, q axis current, and speed.

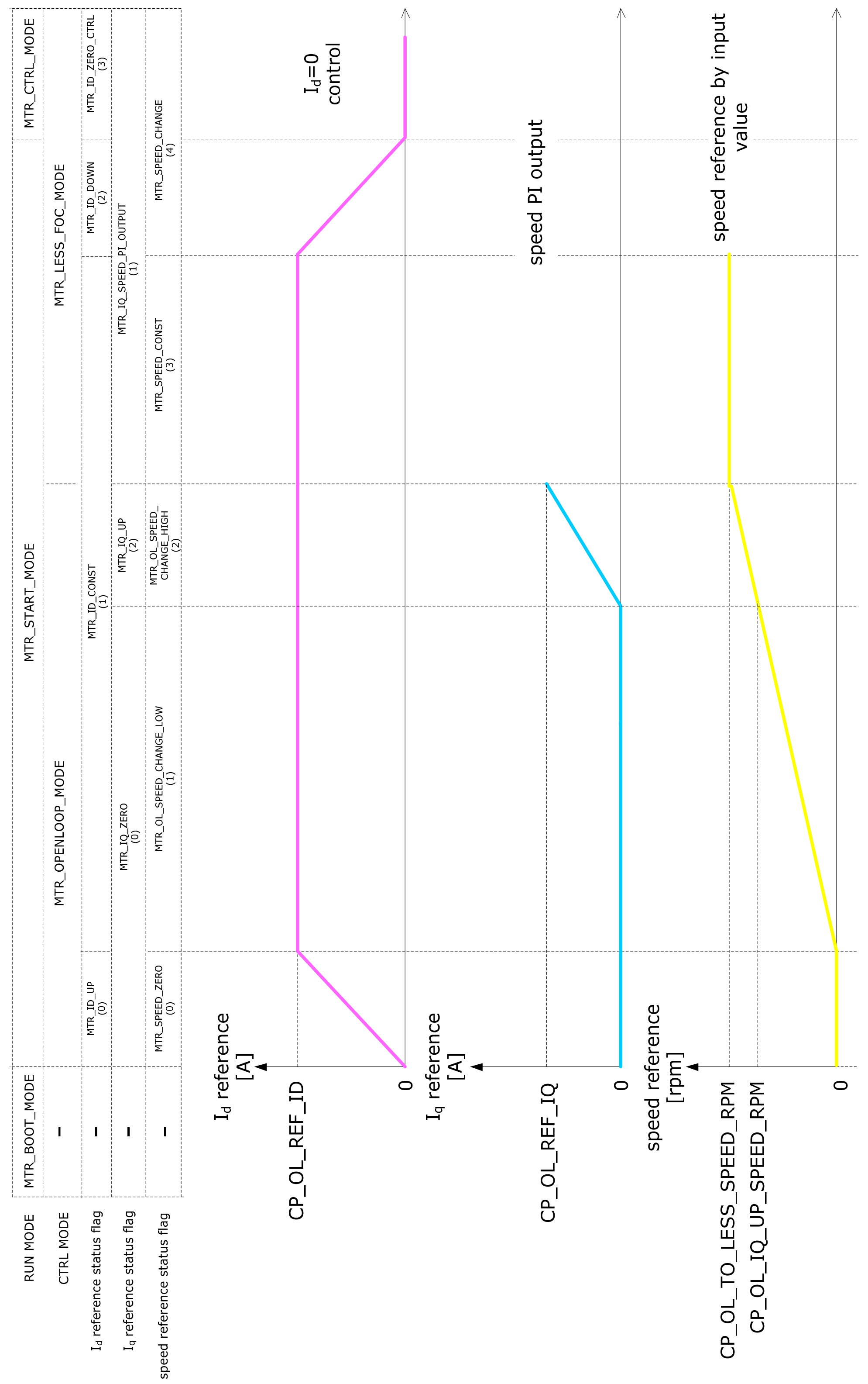

Figure 3-4 Startup Control of Sensorless Vector Control Software (Target Software (1), (2))

For details of the sensorless vector control, refer to the application note 'Vector control of permanent magnetic synchronous motor: algorithm'.

3.1.5 System protection function

This control program has the following five types of error status and executes emergency stop functions in case of occurrence of respective errors. Table 3-5 shows each setting value for the system protection function.

・Over current error

High impedance output is made to the PWM output port in response to an emergency stop signal (over current detection) from the hardware. In addition, U, V, and W phase currents are monitored in over current monitoring cycle. When an over current (when the current exceeds the over current limit value) is detected, the CPU executes emergency stop (software detection).

・Over voltage error

The inverter bus voltage is monitored in over voltage monitoring cycle. When an over voltage is detected (when the voltage exceeds the over voltage limit value), the CPU performs emergency stop. Here, the over voltage limit value is set in consideration for the error of resistance value and error of supply voltage by AC adapter etc.

・Low voltage error

The inverter bus voltage is monitored in low-voltage monitoring cycle. The CPU performs emergency stop when low voltage (when voltage falls below the limit value) is detected.

・Over speed error

The rotation speed is monitored in rotation speed monitoring cycle. The CPU performs emergency stop when the speed is over the limit value.

・IPM temperature error

The IPM temperature is monitored by IPM temperature monitoring cycle. The CPU performs emergency stop when high temperature is detected (when it exceeds the IPM temperature limit value).

Table 3-5 Setting Values of the System Protection Function

		(1)	(2)
Over current error	Over current limit value [A]	4.8	4.8
	Monitoring cycle [μs]	100	100
Over voltage error	Over voltage limit value [V]	28	400
	Monitoring cycle [μs]	100	100
Low voltage error	Low voltage limit value [V]	0	100
	Monitoring cycle [μs]	100	100
Over speed error	Speed limit value [rpm]	4000	6000
	Monitoring cycle [μs]	100	100
IPM temperature error	High temperature limit value [V]	-	3
	Monitoring cycle [μs]	-	100

3.2 Function specifications of sensorless vector control software

Multiple control functions are used in this control program. Lists of control functions are given below.

For detailed processing, please refer to flowcharts or source files.

Table 3-6 List of Control Functions [1/6]

File name	Function name	Process overview
main.c	main Input: None Output: None	・Hardware initialization function call ・User interface initialization function call ・Initialization function call of the variable used in the main process ・Status transition and event execution function call ・Main process ⇒ Main process execution function call ⇒ Watchdog timer clear function call
	ics_ui Input: None Output: None	ICS user interface use
	ctrl_ui (target software: (1)) Input: None Output: None	・Motor status change ・Determination of rotation speed command value
	software_init Input: None Output: None	Initialization of variables used in the main process
(1) mtr_ctrl_t2001.c (2) mtr_ctrl_t1102.c	R_MTR_ChargeCapacitor (target software: (2)) Input: None Output: None	Wait for capacity charging time
	ic_gate_on (target software: (2)) Input: None Output: None	Switching gate signal for inrush current prevention ON
	get_vr1 (target software: (1)) Input: None Output: (uint16) ad_data / A/D conversion result	VR1 status acquisition
	get_sw1 (target software: (1)) Input: None Output: (uint8) tmp_port / SW1 level	SW1 status acquisition
	get_sw2 (target software: (1)) Input: None Output: (uint8) tmp_port / SW2 level	SW2 status acquisition
	led1_on Input: None Output: None	Turning LED1 ON
	led2_on Input: None Output: None	Turning LED2 ON
	led1_off Input: None Output: None	Turning LED1 OFF
	led2_off Input: None Output: None	Turning LED2 OFF

Table 3-7 List of Control Functions [2/6]

File name	Function name	Process overview
mtr_ctrl_rx62t100.c	R_MTR_InitHardware Input: None Output: None	Initialization of the clock and peripheral functions
	mtr_init_cmt Input: None Output: None	Initialization of CMT
	mtr_init_poe3 Input: None Output: None	Initialization of POE3
	init_wdt Input: None Output: None	Initialization of the watchdog timer
	clear_wdt Input: None Output: None	Clearing the watchdog timer
	mtr_clear_oc_flag Input: None Output: None	Clearing the high impedance state
	mtr_clear_cmt0_flag Input: None Output: None	Clearing the interrupt flag

Table 3-8 List of Control Functions [3/6]

File name	Function name	Process overview
mtr_interrupt.c	mtr_over_current_interrupt Input: None Output: None	Overcurrent detection process ・Event processing selection function call ・Changing the motor status ・High impedance state clearing function call
	mtr_mtu4_interrupt Input: None Output: None	Calling every 100 [μs] ・Vector control ・Current PI control
	mtr_cmt0_interrupt Input: None Output: None	Calling every 1 [ms] ・Start control ・Speed PI control

Table 3-9 List of Control Functions [4/6]

File name	Function name	Process overview
mtr_spm_less_foc.c	R_MTR_InitSequence Input: None Output: None	Initialization of the sequence process
	R_MTR_ExecEvent Input: (uint8)u1_event / occurred event Output: None	• Changing the status • Calling an appropriate process execution function for the occurred event
	mtr_act_run Input: (uint8)u1_state / motor status Output: (uint8)u1_state / motor status	• Variable initialization function call upon motor startup • Motor control startup function call
	mtr_act_stop Input: (uint8)u1_state / motor status Output: (uint8)u1_state / motor status	Motor control stop function call
	mtr_act_none Input: (uint8)u1_state / motor status Output: (uint8)u1_state / motor status	No processing is performed.
	mtr_act_reset Input: (uint8)u1_state / motor status Output: (uint8)u1_state / motor status	Global variable initialization
	mtr_act_error Input: (uint8)u1_state / motor status Output: (uint8)u1_state / motor status	Motor control stop function call
	mtr_start_init Input: None Output: None	Initializing only the variables required for motor startup
	mtr_angle_speed Input: None Output: None	Position and speed calculation processing
	mtr_pi_ctrl Input: MTR_PI_CTRL *pi_ctrl/ PI control structure Output: (float32)f4_ref / PI control output value	Current PI control
	mtr_set_variables Input: None Output: None	Setting motor variables
	R_MTR_IcsInput Input: MTR_ICS_INPUT *ics_input / structure for ICS Output: None	Setting the buffer
	R_MTR_SetSpeed Input: (int16)ref_speed / speed command value Output: None	Setting the speed command value
	R_MTR_SetDir Input: uint8 dir/ rotation direction Output: None	Setting the rotation direction
	R_MTR_GetSpeed Input: None Output: (int16) s2_speed_rpm / speed	Obtaining the speed calculation value
	R_MTR_GetDir Input: None Output: (uint8) g_u1_direction / rotation direction	Obtaining the rotation direction
	R_MTR_GetStatus Input: None Output: (uint8)g_u1_mode_system / motor status	Obtaining the motor status

Table 3-10 List of Control Functions [5/6]

File name	Function name	Process overview
mtr_spm_less_foc.c	mtr_error_check Input: None Output: None	Error monitoring and detection
	mtr_set_speed_ref Input: None Output: None	Setting the command value for speed control
	mtr_set_iq_ref Input: None Output: None	Setting the q axis current command value
	mtr_set_id_ref Input: None Output: None	Setting the d axis current command value
	mtr_calc_mod Input: float32 f4_vu / U phase voltage float32 f4_vv / V phase voltage float32 f4_vw / W phase voltage float32 f4_vdc/ bus voltage Output: None	Modulation factor calculation

Table 3-11 List of Control Functions [6/6]

File name	Function name	Process overview
(1) mtr_ctrl_rx62t100_t2001.c (2) mtr_ctrl_rx62t100_t1102.c	mtr_init_mtu Input: None Output: None	Initial setting of MTU3
	mtr_init_io_port Input: None Output: None	Initial setting of IO ports
	mtr_init_ad_converter Input: None Output: None	Initial setting of the A/D converter
	init_ui Input: None Output: None	Initialization of UI
	mtr_ctrl_start Input: None Output: None	Motor startup processing
	mtr_ctrl_stop Input: None Output: None	Motor stop processing
	mtr_get_vr1 (target software: (1)) Input: None Output: (unit16)u2_temp/ VR1 A/D conversion value	VR1 A/D conversion
	mtr_get_iuiwvdc Input: (float32) *f4_iu_ad / U phase current A/D conversion value : (float32) *f4_iw_ad / W phase current A/D conversion value : (float32) *f4_vdc_ad / Vdc A/D conversion value Output: None	A/D conversion of U phase current, W phase current, and inverter bus voltage
	mtr_get_ipm_temperature (target software: (2)) Input: None Output: (int16) s2_temp / IPM temperature A/D conversion value	IPM temperature A/D conversion
	mtr_clear_mtu4_flag Input: None Output: None	Clearing the interrupt flag
	mtr_inv_set_uvw Input: float32 f4_modu,/ U phase modulation factor : float32 f4_modv, / V phase modulation factor : float32 f4_modw / W phase modulation factor Output: None	PWM output setting
	mtr_init_register Input: None Output: None	Initialization of buffer register

3.3 List of sensorless vector control software variables

Lists of variables used in this control program are given below. Note that the local variables are not mentioned.

Table 3-12 List of Variables [1/4]

Variable name	Type	Content	Remarks
g_s2_max_speed (target software: (1))	int16	Maximum speed value	Mechanical angle [rpm]
g_s2_ref_speed (target software: (1))	int16	Speed command value	Mechanical angle [rpm]
g_u1_motor_status	uint8	Motor status	
g_u1_reset_req (target software: (1))	uint8	Reset request flag	0: Turning SW2 ON in error status 1: Turning SW2 OFF in error status
g_u1_sw1_cnt (target software: (1))	uint8	SW1 determining counter	Chattering removal
g_u1_sw2_cnt (target software: (1))	uint8	SW2 determining counter	Chattering removal
g_s2_sw_userif (target software: (1))	int16	User interface switch	0: ICS user interface use (default) 1: Board user interface use
g_u1_mode_system	unit8	State management	0: Stop mode 1: Run mode 2: Error mode
g_u2_run_mode	unit16	Operation mode management	0: Boot mode 1: Start mode 2: Control mode
g_u2_ctrl_mode	unit16	Control mode	1: Open loop mode 5: Sensorless vector control mode
g_u1_error_status	unit8	Error status management	1: Over current error 2: Over voltage error 3: Rotation speed error 7: Low voltage error 8: Over IPM temperature error (target software : (2)) 0xFF: Undefined error
g_f4_vdc_ad	float32	Inverter bus voltage	[V]
g_f4_vd_ref	float32	d axis output voltage command value	Current PI control output value [V]
g_f4_vq_ref	float32	q axis output voltage command value	Current PI control output value [V]
g_f4_iu_ad	float32	U phase current	[A]
g_f4_pre_iu_ad	float32	Previous value of U phase current	[A]
g_f4_iv_ad	float32	V phase current	[A]
g_f4_iw_ad	float32	W phase current	[A]
g_f4_pre_iw_ad	float32	Previous value of W phase current	[A]
g_f4_offset_iu	float32	U phase current offset value	[A]
g_f4_offset_iw	float32	W phase current offset value	[A]
g_f4_id_lpf	float32	d axis current	[A]
g_f4_iq_lpf	float32	q axis current	[A]
g_f4_pre_id_lpf	float32	Previous value of d axis current	[A]
g_f4_pre_iq_lpf	float32	Previous value of q axis current	[A]
g_f4_kp_id	float32	d axis current PI control proportional term gain	
g_f4_ki_id	float32	d axis current PI control integral term gain	

Table 3-13 List of Variables [2/4]

Variable name	Type	Content	Remarks
g_f4_kp_iq	float32	q axis current PI control proportional term gain	
g_f4_ki_iq	float32	q axis current PI control integral term gain	
g_f4_lim_vd	float32	d axis current PI control output limit value	[V]
g_f4_lim_vq	float32	q axis current PI control output limit value	[V]
g_f4_lim_iq	float32	Speed PI control output limit value	[A]
g_f4_kp_speed	float32	Speed PI control proportional term gain	
g_f4_ki_speed	float32	Speed PI control integral term gain	
g_f4_ilim_vd	float32	d axis current PI control integral term limit value	[V]
g_f4_ilim_vq	float32	q axis current PI control integral term limit value	[V]
g_f4_ilim_iq	float32	Speed PI control integral term limit value	[A]
g_f4_id_ref	float32	d axis current command value	[A]
g_f4_iq_ref	float32	q axis current command value	Speed PI control output value [A]
g_f4_speed_rad	float32	Speed operation value	Electrical angle [rad/s]
g_f4_ref_speed_rad_pi	float32	Command value for speed PI control	Electrical angle [rad/s]
g_f4_angle_rad	float32	Rotor position	[rad]
g_f4_max_speed_rad	float32	Maximum speed command value	[rad/s]
g_f4_refu	float32	U phase voltage command value	[V]
g_f4_refv	float32	V phase voltage command value	[V]
g_f4_refw	float32	W phase voltage command value	[V]
g_f4_inv_limit	float32	Phase voltage limit value	[V]
g_f4_speed_lpf_k	float32	Speed LPF parameter	
g_f4_current_lpf_k	float32	Current LPF parameter	
g_f4_offset_lpf_k	float32	LPF parameter of current offset value	
vd	MTR_PI_CTRL	Structure for d axis current PI control	
vq	MTR_PI_CTRL	Structure for q axis current PI control	
speed	MTR_PI_CTRL	Structure for speed PI control	
mtr_p	MTR_ PARAMETER	Motor parameter and control parameter	
g_u1_direction	unit8	Rotation direction	0: CW 1: CCW
g_u1_dir_buff	unit8	Command rotation direction	0: CW 1: CCW
g_u1_enable_write	unit8	Variable for ICS UI	
ics_input	MTR_ICS_INPUT	Structure for ICS UI	
g_u2_cnt_adjust	unit16	Counter to calculate current offset	
g_u1_flag_id_ref	unit8	d axis current command value management flag	0: d axis current increase 1: Constant d axis current 2: d axis current decrease 3: d axis current 0
g_u1_flag_iq_ref	unit8	q axis current command value management flag	0: q axis current 0 1: Speed PI output 2: q axis current increase 3: q axis current decrease
g_f4_temp_speed_rad	float32	Variable to store speed	Electrical angle [rad/s]
g_f4_temp_ref_speed_rad	float32	Variable to store speed command value	Electrical angle [rad/s]
g_u1_flag_down_to_ol	unit8	Open loop mode transition flag	0: Stay 1: Transition

Table 3-14 List of Variables (3/4)

Variable name	Type	Content	Remarks
g_u1_flag_offset_calc	unit8	Current offset value calculation flag	0: Calculation in transition to the boot mode 1: Calculation in transition to the boot mode (first time only)
g_f4_iq_down_step	float32	q axis current subtraction value	[A]
g_f4_emf_est	float32	Estimation value of inductive voltage	[V]
g_f4_emf_calc	float32	Calculation value of inductive voltage	[V]
g_f4_id_ref_buff	float32	Variable to store d axis current command value	[A]
g_f4_iq_ref_buff	float32	Variable to store q axis current command value	[A]
g_u2_cnt_speed_const	uint16	Counter of reference speed constant time	
g_u1_flag_speed_ref	uint8	Speed command value management flag	0: Speed 0 1: At the time of low-speed open loop 2: At the time of high-speed open loop 3: Constant speed 4: Variable speed
g_f4_i_gamma	float32	γ axis current	[A]
g_f4_i_delta	float32	δ axis current	[A]
g_f4_di_gamma	float32	γ axis current error	[A]
g_f4_di_delta	float32	δ axis current error	[A]
g_f4_k_emf	float32	Speed electromotive force estimation gain	
g_f4_k_theta	float32	Position estimation gain	
g_f4_tdspeed_lpf	float32	Control cycle × difference in speed	
g_f4_pre_speed_rad	float32	Previous speed value	Electrical angle [rad/s]
g_f4_ol_to_less_speed_rad	float32	Sensorless control switching speed	Electrical angle [rad/s]
g_f4_ol_iq_up_speed_rad	float32	Speed at start of q axis current command value increase	Electrical angle [rad/s]
g_f4_ol_iq_up_step	float32	Reference q axis current adding value in open loop mode	[A]
g_f4_id_down_step	float32	Reference d axis current subtracting value	[A]
g_f4_ref_speed_const_time	float32	Time of constant speed command value	[ms]
g_f4_fluctuation_limit	float32	Limit value of speed fluctuation	[rad/s]
g_f4_ol_iq_down_step	float32	Reference q axis current subtracting value in open loop mode	[A]
g_f4_ol_id_up_step	float32	Reference d axis current adding value in open loop mode	[A]
g_f4_ol_ref_id	float32	d axis current command value in open loop mode	[A]
g_f4_ol_ref_iq	float32	q axis current command value in open loop mode	[A]
ics_input_buff	MTR_ICS_INPUT	Structure for ICS UI	
g_s2_mode_system	int16	State management	
g_s2_enable_write	int16	Variable for ICS UI	
g_u1_cnt_ics	uint8	ICS decimation counter	
g_f4_ref_speed_rad	float32	Reference speed	Electrical angle [rad/s]
g_f4_less_to_ol_speed_rad	float32	Openloop control switching speed	Electrical angle [rad/s]

Table 3-15 List of Variables (4/4)

Variable name	Type	Content	Remarks
g_f4_offset_calc_time	float32	Calculation time for current offset	[ms]
g_f4_accel	float32	Acceleration	[rad/s^2]
g_f4_modu	float32	U phase modulation factor	
g_f4_modv	float32	V phase modulation factor	
g_f4_modw	float32	W phase modulation factor	
g_f4_ipm_temperature_ad	float32	IPM temperature voltage conversion value	[V]

3.4 List of sensorless vector control software structures

Lists of structures used in this control program are given below.

Table 3-16 List of structures

	Member	Type	Content	Remarks
MTR_PI_CTRL	f4_err	float32	Error	
	f4_kp	float32	PI control proportional gain	
	f4_ki	float32	PI control integral gain	
	f4_limit	float32	PI control output limit value	
	f4_refi	float32	Integral term output value	
	f4_ilimit	float32	Integral term output limit value	
MTR_PARAMETER	f4_mtr_r	float32	Resistance	[Ω]
	f4_mtr_l	float32	Inductance	[H]
	f4_mtr_m	float32	Magnet flux	[Wb]
	f4_mtr_t	float32	Control period	[s]
	f4_mtr_t_l	float32	f4_mtr_t / f4_mtr_l	
	f4_mtr_t_m	float32	f4_mtr_t / f4_mtr_m	
MTR_ICS_INPUT	s2_ref_speed	int16	Reference speed	Mechanical angle [rpm]
	s2_direction	int16	Rotation direction	0 : CW / 1 : CCW
	f4_kp_speed	float32	Speed PI control proportional gain	
	f4_ki_speed	float32	Speed PI control Integral gain	
	f4_kp_id	float32	d-axis current PI control proportional gain	
	f4_ki_id	float32	d-axis current PI control integral gain	
	f4_kp_iq	float32	q-axis current PI control proportional gain	
	f4_ki_iq	float32	q-axis current PI control integral gain	
	f4_speed_lpf_k	float32	Speed LPF parameter	
	f4_current_lpf_k	float32	Current LPF parameter	
	f4_mtr_r	float32	Resistance	[Ω]
	f4_mtr_l	float32	Inductance	[H]
	f4_mtr_m	float32	Magnet flux	[Wb]
	f4_offset_lpf_k	float32	Current offset LPF parameter	
	s2_max_speed	int16	Maximum speed	Mechanical angle [rpm]
	s2_ol_to_less_speed	int16	Sensorless control switching speed	Mechanical angle [rpm]
	s2_ol_iq_up_speed	int16	Speed at start of increasing reference q axis current	Mechanical angle [rpm]
	s2_less_to_ol_speed	int16	Openloop control switching speed	Mechanical angle [rpm]
	f4_ol_ref_id	float32	Reference d axis current	[A]
	f4_ol_id_up_time	float32	Reference d axis current adding time in openloop mode	[ms]
	f4_id_down_time	float32	Reference d axis current subtracting time	[ms]
	f4_ref_speed_const_time	float32	Time of constant speed command value	[ms]
	f4_accel	float32	Acceleration	[rad/s^2]
	f4_fluctuation_limit	float32	Limit value of speed fluctuation	[rad/s]
	f4_ol_iq_down_time	float32	Reference q axis current subtracting time in open loop mode	[ms]
	f4_ol_ref_iq	float32	Reference q axis current	[A]
	f4_ol_iq_up_time	float32	Reference q axis current adding time in open loop mode	[ms]
	f4_offset_calc_time	float32	Calculation time for current offset	[ms]

3.5 Sensorless vector control software macro definitions

Lists of macro definitions used in this control program are given below.

Table 3-17 List of Macro Definitions [1/12]

File name	Macro name	Definition value	Remarks
main.h	ICS_UI	(1): 0 (2): Undefined	ICS user interface use
	BOARD_UI	(1): 1 (2): Undefined	Board user interface use
	MAX_SPEED	CP_MAX_SPEED_RPM	Rotation speed command maximum value (mechanical angle) [rpm]
	OL_TO_LESS_SPEED_RPM	CP_OL_TO_LESS_SPEED_RPM	Rotation speed command minimum value (mechanical angle) [rpm]
	ID_PI_KP	CP_ID_PI_KP	d axis current PI control proportional term gain
	ID_PI_KI	CP_ID_PI_KI	d axis current PI control integral term gain
	IQ_PI_KP	CP_IQ_PI_KP	q axis current PI control proportional term gain
	IQ_PI_KI	CP_IQ_PI_KI	q axis current PI control integral term gain
	SPEED_PI_KP	CP_SPEED_PI_KP	Speed PI control proportional term gain
	SPEED_PI_KI	CP_SPEED_PI_KI	Speed PI control integral term gain
	EMF_EST_K	CP_EMF_EST_K	Speed electro-motive force estimation gain
	THETA_EST_K	CP_THETA_EST_K	Position estimation gain
	SPEED_LPF_K	CP_SPEED_LPF_K	Speed LPF parameter
	CURRENT_LPF_K	CP_CURRENT_LPF_K	Current LPF parameter
	MAGNETIC_FLUX	MP_MAGNETIC_FLUX	Flux [Wb]
	RESISTANCE	MP_RESISTANCE	Resistance [Ω]
	INDUCTANCE	MP_INDUCTANCE	Inductance [H]
	OFFSET_LPF_K	CP_OFFSET_LPF_K	LPF parameter of current offset value
	OL_IQ_UP_SPEED	CP_OL_IQ_UP_SPEED_RPM	Speed at start of q axis current command value increase (mechanical angle) [rpm]
	LESS_TO_OL_SPEED	CP_LESS_TO_OL_SPEED_RPM	Open loop switching speed (mechanical angle) [rpm]
	OL_REF_ID	CP_OL_REF_ID	Command d axis current in open loop mode [A]
	OL_ID_UP_TIME	CP_OL_ID_UP_TIME	Command d axis current adding time in open loop mode [ms]
	ID_DOWN_TIME	CP_ID_DOWN_TIME	Command d axis current subtracting time [ms]
	REF_SPEED_CONST_TIME	CP_REF_SPEED_CONST_TIME	Time during which speed command value is constant [ms]
	ACCEL_MODE0	CP_ACCEL_MODE0	Acceleration in start mode [rad/s^2]
	ACCEL_MODE1	CP_ACCEL_MODE1	Acceleration in control mode [rad/s^2]

Table 3-18 List of Macro Definitions [2/12]

File name	Macro name	Definition value	Remarks
main.h	FLUCTUATION_LIMIT	CP_FLUCTUATION_LIMIT	Speed fluctuation limit value [rad/s]
	OL_IQ_DOWN_TIME	CP_OL_IQ_DOWN_TIME	Command q axis current subtracting time [ms]
	OL_REF_IQ	CP_OL_REF_IQ	q axis current command value in open loop mode [A]
	OL_IQ_UP_TIME	CP_OL_IQ_UP_TIME	q axis current command value adding time in open loop mode [ms]
	OFFSET_CALC_TIME	CP_OFFSET_CALC_TIME	Current offset value calculation time [ms]
	SW_ON	(1): 0 (2): Undefined	Active in case of “Low”
	SW_OFF	(1): 1 (2): Undefined	
	CHATTERING_CNT	(1): 10 (2): Undefined	Chattering removal
	VR1_SCALING	(1): 4 (2): Undefined	Constant for creating rotation speed command value
	ADJUST_OFFSET	(1): 0x1FF (2): Undefined	
	POLE_PAIRS	MP_POLE_PAIRS	Constant for correcting number of pole pairs
	M_CW	0	Rotation direction
	M_CCW	1	
	ICS_INT_LEVEL	6	Interrupt priority level for ICS

Table 3-19 List of Macro Definitions [3/12]

File name	Macro name	Definition value	Remarks
motor_parameter.h	MP_POLE_PAIRS	(1): 7 (2): 5	Number of pole pairs
	MP_MAGNETIC_FLUX	(1): 0.006198f (2): 0.091f	Flux [Wb]
	MP_RESISTANCE	(1): 0.453f (2): 1.69235f	Resistance [Ω]
	MP_INDUCTANCE	(1): 0.0009477f (2): 0.00889f	Inductance [H]

Table 3-20 List of Macro Definitions [4/12]

File name	Macro name	Definition value	Remarks
(1) mtr_ctrl_rx62t100_t2001.h (2) mtr_ctrl_rx62t100_t1102.h	MTR_PWM_TIMER_FREQ	96.0f	PWM timer count frequency [MHz]
	MTR_CARRIER_FREQ	20.0f	Carrier frequency [kHz]
	MTR_DEADTIME	(1): 2 (2): 3	Dead time [μs]
	MTR_DEADTIME_SET	(uint16)(MTR_DEADTIME MTR_PWM_TIMER_FREQ)	Dead time setting value
	MTR_AD_FREQ	48.0f	Frequency of A/D conversion clock
	MTR_AD_SAMPLING_CYCLE	26.0f	A/D sampling time [Cycle]
	MTR_AD_SAMPLING_TIME	MTR_AD_SAMPLING_CYCLE / MTR_AD_FREQ	A/D sampling time [μs]
	MTR_AD_TIME_SET	(uint16)(MTR_PWM_TIMER_FREQ * MTR_AD_SAMPLING_TIME)	A/D sampling time count value
	MTR_CARRIER_SET	(uint16)((MTR_PWM_TIMER_FREQ * 1000 / MTR_CARRIER_FREQ / 2)+ MTR_DEADTIME_SET)	Carrier setting value
	MTR_HALF_CARRIER_SET	(uint16)(MTR_CARRIER_SET / 2)	Half of “MTR_CARRIER_SET”

Table 3-21 List of Macro Definitions [5/12]

File name	Macro name	Definition value	Remarks
(1) mtr_ctrl_rx62t100_t2001.h (2) mtr_ctrl_rx62t100_t1102.h	MTR_PORT_UP	PORT7.DR.BIT.B1	U phase (positive phase) output port
	MTR_PORT_UN	PORT7.DR.BIT.B4	U phase (negative phase) output port
	MTR_PORT_VP	PORT7.DR.BIT.B2	V phase (positive phase) output port
	MTR_PORT_VN	PORT7.DR.BIT.B5	V phase (negative phase) output port
	MTR_PORT_WP	PORT7.DR.BIT.B3	W phase (positive phase) output port
	MTR_PORT_WN	PORT7.DR.BIT.B6	W phase (negative phase) output port
	MTR_PORT_SW1	(1): PORT9.PORT.BIT.B1 (2): Undefined	SW1 input port
	MTR_PORT_SW2	(1): PORT9.PORT.BIT.B2 (2): Undefined	SW2 input port
	MTR_PORT_LED1	PORTA.DR.BIT.B2	LED1 output port
	MTR_PORT_LED2	PORTA.DR.BIT.B3	LED2 output port
	MTR_LED_ON	0	Active in case of “Low”
	MTR_LED_OFF	1	
	MTR_INPUT_V	(1): 24 (2): 240*1.41421356	Power supply voltage [V]
	MTR_IC_GATE_ON_V	(1): Undefined (2): (int32)(MTR_INPUT_V*0.8f)	Power supply voltage * 80 % [V]
	MTR_HALF_VDC	MTR_INPUT_V/2.0f	Power supply voltage/2 [V]
	MTR_ADC_SCALING	0x7FF	Constant for adjusting ADC offset
	MTR_CURRENT_SCALING	(1): 20.0f/4095.0f (2): 100.0f/4095.0f	Current A/D conversion value resolution
	MTR_VDC_SCALING	(1): 111.0f/4095.0f (2): 686.8f/4095.0f	Inverter bus voltage A/D conversion value resolution
	MTR_IPMTEMPERATURE_SCALING	(1): Undefined (2): 5.0f/1023	IPM temperature A/D conversion value resolution
	MTR_OVERCURRENT_LIMIT	4.8f	Current limit value [A]
	MTR_OVERVOLTAGE_LIMIT	(1): 28 (2): 400	High voltage limit value [V]
	MTR_UNDERVOLTAGE_LIMIT	(1): 0 (2): 100	Low voltage limit value [V]
	MTR_OVERIPMTEMPERATURE_LIMIT	(1): Undefined (2): 3	IPM temperature limit value [V]

Table 3-22 List of Macro Definitions [6/12]

File name	Macro name	Definition value	Remarks
(1) mtr_ctrl_rx62t100_t2001.h (2) mtr_ctrl_rx62t100_t1102.h	MTR_PORT_IC_GATE	(1): Undefined (2): PORTB.DR.BIT.B3	Port for inrush current prevention circuit
	MTR_IC_GATE_ON	(1): Undefined (2): 1	

Table 3-23 List of Macro Definitions [7/12]

File name	Macro name	Definition value	Remarks
mtr_spm_less_foc.h	MTR_INT_DECIMATION	1	Number of interrupt decimation times
	MTR_CTRL_PERIOD	(MTR_INT_DECIMATION + 1)/ (MTR_CARRIER_FREQ*1000)	Control cycle [s]
	MTR_CONTROL_FREQ	(MTR_CARRIER_FREQ*1000)/ (MTR_INT_DECIMATION + 1)	Control frequency [Hz]
	MTR_M	MP_MAGNETIC_FLUX	Flux [Wb]
	MTR_R	MP_RESISTANCE	Resistance [Ω]
	MTR_L	MP_INDUCTANCE	Inductance [H]
	MTR_T_L	MTR_CTRL_PERIOD/MTR_L	T/L [s/H]
	MTR_T_M	MTR_CTRL_PERIOD/MTR_M	T/M [s/Wb]
	MTR_POLE_PAIRS	MP_POLE_PAIRS	Number of pole pairs
	MTR_SPEED_LIMIT_RPM	(1): 4000 (2): 6000	Speed limit value (mechanical angle) [rpm]
	MTR_SPEED_LIMIT_RAD	MTR_SPEED_LIMIT_RPM*MTR_RPM_RAD*MTR_POLE_PAIRS	Speed limit value (electrical angle) [rad/s]
	MTR_TWOPI	2*3.14159265f	2π
	MTR_TWOPI_3	MTR_TWOPI/3	$2\pi/3$
	MTR_SQRT_2	1.41421356f	$\sqrt{2}$
	MTR_SQRT_3	1.7320508f	$\sqrt{3}$
	MTR_SQRT_2_3	0.81649658f	$\sqrt{(2/3)}$
	MTR_RPM_RAD	MTR_TWOPI/60	$2\pi/60$
	MTR_ID_PI_KP	CP_ID_PI_KP	d axis current PI control proportional term gain
	MTR_ID_PI_KI	CP_ID_PI_KI	d axis current PI control integral term gain
	MTR_IQ_PI_KP	CP_IQ_PI_KP	q axis current PI control proportional term gain
	MTR_IQ_PI_KI	CP_IQ_PI_KI	q axis current PI control integral term gain
	MTR_SPEED_PI_KP	CP_SPEED_PI_KP	Speed PI control proportional term gain
	MTR_SPEED_PI_KI	CP_SPEED_PI_KI	Speed PI control integral term gain
	MTR_EMF_EST_K	CP_EMF_EST_K	Speed electro-motive force estimation gain
	MTR_THETA_EST_K	CP_THETA_EST_K	Position estimation gain
	MTR_SPEED_LPF_K	CP_SPEED_LPF_K	Speed LPF parameter
	MTR_CURRENT_LPF_K	CP_CURRENT_LPF_K	Current LPF parameter
	MTR_OFFSET_LPF_K	CP_OFFSET_LPF_K	LPF parameter of current offset value
	MTR_LIMIT_IQ	4.5	Speed PI control output limit value [A]
	MTR_I_LIMIT_IQ	4.5	Speed PI control integral term limit value [A]
	MTR_LIMIT_VD	(1): 12 (2): 120*1.41421356	d axis current PI control output limit value [V]

Table 3-24 List of Macro Definitions [8/12]

File name	Macro name	Definition value	Remarks
mtr_spm_less_foc.h	MTR_LIMIT_VQ	(1): 12 (2): 120*1.41421356	q axis current PI control output limit value [V]
	MTR_I_LIMIT_VD	(1): 12 (2): 120*1.41421356	d axis current PI control integral term limit value [V]
	MTR_I_LIMIT_VQ	(1): 12 (2): 120*1.41421356	q axis current PI control proportional term limit value [V]
	MTR_MAX_SPEED_RPM	CP_MAX_SPEED_RPM	Maximum speed (mechanical angle) [rpm]
	MTR_MAX_SPEED_RAD	MTR_MAX_SPEED_RPM *MTR_POLE_PAIRS*MTR_TWOPI/60	Maximum speed (electrical angle) [rad/s]
	MTR_OL_TO_LESS_SPEED_RPM	CP_OL_TO_LESS_SPEED_RPM	Sensorless control switching speed (mechanical angle) [rpm]
	MTR_OL_TO_LESS_SPEED_RAD	MTR_OL_TO_LESS_SPEED_RPM*MTR_POLE_PAIRS*MTR_TWOPI/60	Sensorless control switching speed (electrical angle) [rad/s]
	MTR_OL_IQ_UP_SPEED_RPM	CP_OL_IQ_UP_SPEED_RPM	Speed at start of q axis current command value increase (mechanical angle) [rpm]
	MTR_OL_IQ_UP_SPEED_RAD	MTR_OL_IQ_UP_SPEED_RPM*MTR_POLE_PAIRS*MTR_TWOPI/60	Speed at start of q axis current command value increase (electrical angle) [rad/s]
	MTR_LESS_TO_OL_SPEED_RPM	CP_LESS_TO_OL_SPEED_RPM	Open loop switching speed (mechanical angle) [rpm]
	MTR_LESS_TO_OL_SPEED_RAD	MTR_LESS_TO_OL_SPEED_RPM*MTR_POLE_PAIRS*MTR_TWOPI/60	Open loop switching speed (electrical angle) [rad/s]
	MTR_OL_REF_ID	CP_OL_REF_ID	Command d axis current in open loop mode [A]
	MTR_OL_ID_UP_TIME	CP_OL_ID_UP_TIME	Command d axis current adding time in open loop mode [ms]
	MTR_OL_ID_UP_STEP	MTR_OL_REF_ID / MTR_OL_ID_UP_TIME	Command d axis current adding value in open loop mode [A]
	MTR_ID_DOWN_TIME	CP_ID_DOWN_TIME	Command d axis current subtracting time [ms]
	MTR_ID_DOWN_STEP	MTR_OL_REF_ID / MTR_ID_DOWN_TIME	Command d axis current subtracting value [A]
	MTR_REF_SPEED_CONST_TIME	CP_REF_SPEED_CONST_TIME	Time during which speed command value is constant [ms]
	MTR_ACCEL_MODE0	CP_ACCEL_MODE0	Acceleration in start mode [rad/s^2]
	MTR_ACCEL_MODE1	CP_ACCEL_MODE1	Acceleration in control mode [rad/s^2]
	MTR_FLUCTUATION_LIMIT	CP_FLUCTUATION_LIMIT	Speed fluctuation limit value [rad/s]
	MTR_OL_IQ_DOWN_TIME	CP_OL_IQ_DOWN_TIME	Command q axis current subtracting time [ms]
	MTR_OL_IQ_DOWN_STEP	1/ MTR_OL_IQ_DOWN_TIME	Inverse of command q axis current subtracting time
	MTR_OL_REF_IQ	CP_OL_REF_IQ	q axis current command value in open loop mode [A]
	MTR_OL_IQ_UP_TIME	CP_OL_IQ_UP_TIME	q axis current command value adding time in open loop mode [ms]

Table 3-25 List of Macro Definitions [9/12]

File name	Macro name	Definition value	Remarks
mtr_spm_less_foc.h	MTR_OL_IQ_UP_STEP	MTR_OL_REF_IQ / MTR_OL_IQ_UP_TIME	Command q axis current adding value in open loop mode [A]
	MTR_OFFSET_CALC_TIME	CP_OFFSET_CALC_TIME	Current offset value calculation time [ms]
	MTR_EVERY_TIME	0	Calculation when transferring current offset value to boot mode
	MTR_ONE_TIME	1	Calculation when transferring current offset value to boot mode (first time only)
	MTR_CW	0	Rotation direction
	MTR_CCW	1	
	MTR_FLG_CLR	0	Flag management
	MTR_FLG_SET	1	
	MTR_ID_UP	0	d axis current: increase
	MTR_ID_CONST	1	d axis current: constant
	MTR_ID_DOWN	2	d axis current: decrease
	MTR_ID_ZERO_CTRL	3	d axis current 0
	MTR_IQ_ZERO	0	q axis current 0
	MTR_IQ_SPEED_PI_OUTPUT	1	Speed PI output
	MTR_IQ_UP	2	q axis current increase
	MTR_IQ_DOWN	3	q axis current decrease
	MTR_SPEED_ZERO	0	Speed 0
	MTR_OL_SPEED_CHANGE_LOW	1	At the time of low-speed open loop
	MTR_OL_SPEED_CHANGE_HIGH	2	At the time of high-speed open loop
	MTR_SPEED_CONST	3	Constant speed
	MTR_SPEED_CHANGE	4	Variable speed
	MTR_BOOT_MODE	0x00	Boot mode
	MTR_START_MODE	0x01	Start mode
	MTR_CTRL_MODE	0x02	Control mode
	MTR_ZERO_PEC_MODE	0x00	Speed 0 position estimation mode
	MTR_OPENLOOP_MODE	0x01	Open loop mode
	MTR_HALL_120_MODE	0x02	Hall sensor 120-degree operation mode
	MTR_LESS_120_MODE	0x03	BEMF sensorless 120-degree operation mode
	MTR_ENCD_FOC_MODE	0x04	Encoder vector operation mode
	MTR_LESS_FOC_MODE	0x05	Sensorless vector control mode
	MTR_OVER_CURRENT_ERROR	0x01	Over current error
	MTR_OVER_VOLTAGE_ERROR	0x02	Over voltage error
	MTR_OVER_SPEED_ERROR	0x03	Over speed error
	MTR_TIMEOUT_ERROR	0x04	Timeout error
	MTR_UNDER_VOLTAGE_ERROR	0x07	Low voltage error
	MTR_OVER_IPMTEMPERATURE_ERROR	(1): Undefined (2): 0x08	IPM temperature error
	MTR_UNKNOWN_ERROR	0xff	Undefined error

Table 3-26 List of Macro Definitions [10/12]

File name	Macro name	Definition value	Remarks
mtr_spm_less_foc.h	MTR_MODE_STOP	0x00	Stop status
	MTR_MODE_RUN	0x01	Rotating
	MTR_MODE_ERROR	0x02	Error status
	MTR_SIZE_STATE	3	Status counts
	MTR_EVENT_STOP	0x00	Motor stop event
	MTR_EVENT_RUN	0x01	Motor startup event
	MTR_EVENT_ERROR	0x02	Motor error event
	MTR_EVENT_RESET	0x03	Motor reset event
	MTR_SIZE_EVENT	4	Event counts

Table 3-27 List of Macro Definitions [11/12]

File name	Macro name	Definition value	Remarks
control_parameter.h	CP_ID_PI_KP	(1): 3.0 (2): 10.8227f	d axis current PI control proportional term gain
	CP_ID_PI_KI	(1): 0.002 (2): 0.1f	d axis current PI control integral term gain
	CP_IQ_PI_KP	(1): 3.0 (2): 10.8227f	q axis current PI control proportional term gain
	CP_IQ_PI_KI	(1): 0.004 (2): 0.08f	q axis current PI control integral term gain
	CP_SPEED_PI_KP	(1): 0.001 (2): 0.01f	Speed PI control proportional term gain
	CP_SPEED_PI_KI	(1): 0.0005 (2): 0.00003f	Speed PI control integral term gain
	CP_EMF_EST_K	(1): 0.2 (2): 1.935f	Speed electro-motive force estimation gain
	CP_THETA_EST_K	(1): 0.2 (2): 0.0948f	Position estimation gain
	CP_SPEED_LPF_K	(1): 0.04 (2): 0.0136f	Speed LPF parameter
	CP_CURRENT_LPF_K	(1): 1.0 (2): 0.8f	Current LPF parameter
	CP_OFFSET_LPF_K	0.1	LPF parameter of current offset value
	CP_MAX_SPEED_RPM	(1): 2000 (2): 3000.0f	Maximum speed (mechanical angle) [rpm]
	CP_OL_TO_LESS_SPEED_RPM	600.0f	Sensorless control switching speed (mechanical angle) [rpm]

Table 3-28 List of Macro Definitions [12/12]

File name	Macro name	Definition value	Remarks
control_parameter.h	CP_OL_IQ_UP_SPEED_RPM	(1): 450.0f (2): 500.0f	Speed at start of q axis current command value increase (mechanical angle) [rpm]
	CP_LESS_TO_OL_SPEED_RPM	(1): 350.0f (2): 300.0f	Open loop switching speed (mechanical angle) [rpm]
	CP_OL_REF_ID	(1): 1.0f (2): 1.25f	Command d axis current in open loop mode [A]
	CP_OL_ID_UP_TIME	(1): 128.0f (2): 64.0f	Command d axis current adding time in open loop mode [ms]
	CP_ID_DOWN_TIME	(1): 512.0f (2): 1024.0f	Command d axis current subtracting time [ms]
	CP_REF_SPEED_CONST_TIME	256.0f	Time during which speed command value is constant [ms]
	CP_ACCEL_MODE0	(1): 0.2 (2): 0.5f	Acceleration in start mode [rad/s^2]
	CP_ACCEL_MODE1	(1): 0.2 (2): 0.5f	Acceleration in control mode [rad/s^2]
	CP_FLUCTUATION_LIMIT	(1): 20.0f (2): 50.0f	Speed fluctuation limit value [rad/s]
	CP_OL_IQ_DOWN_TIME	100.0f	Command q axis current subtracting time [ms]
	CP_OL_REF_IQ	0.6f	q axis current command value in open loop mode [A]
	CP_OL_IQ_UP_TIME	512.0f	q axis current command value adding time in open loop mode [ms]
	CP_OFFSET_CALC_TIME	256	Current offset value calculation time [ms]

3.6 Control flowcharts

3.6.1 Main process

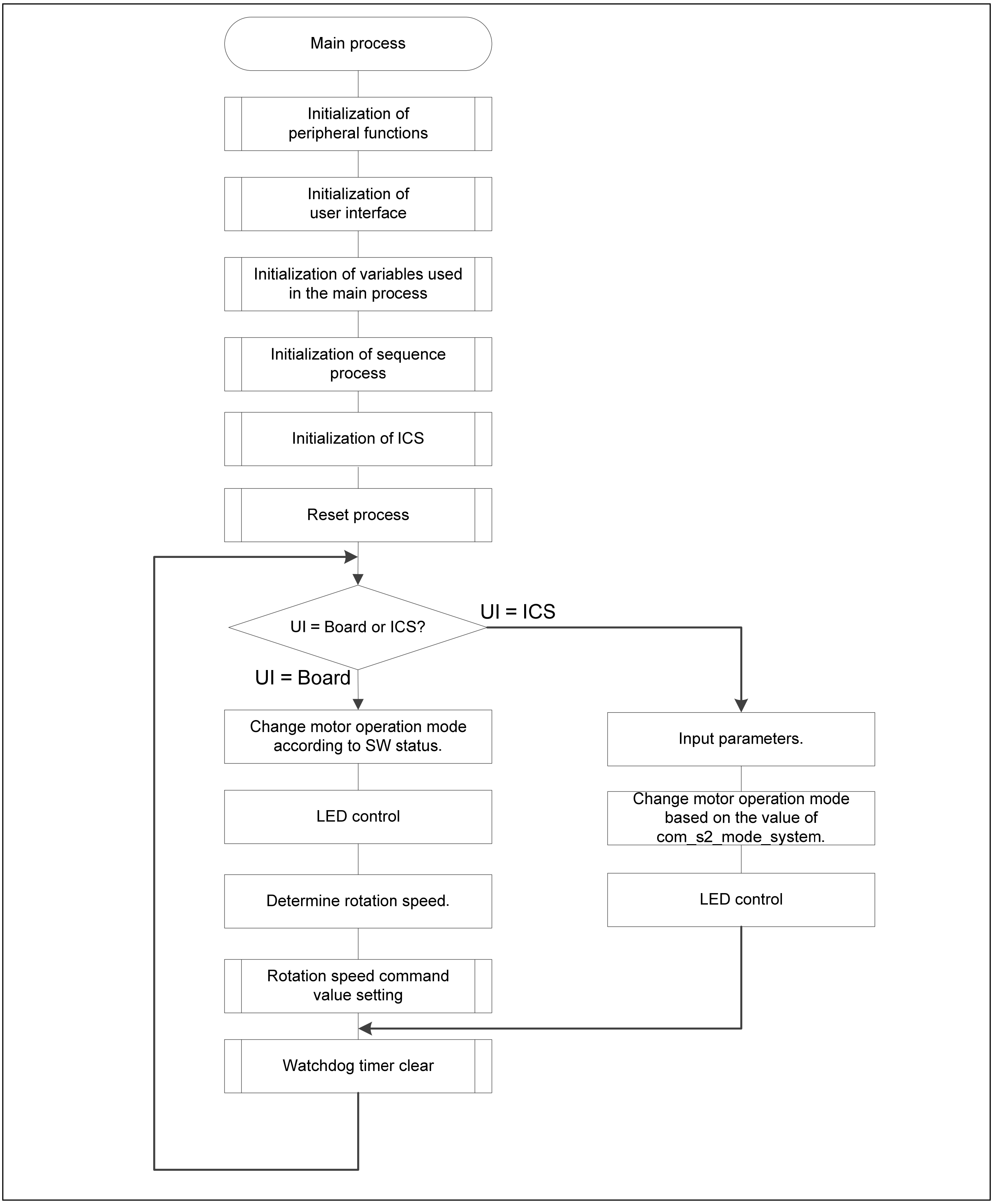

Figure 3-5 Main Process Flowchart (Target Software: (1))

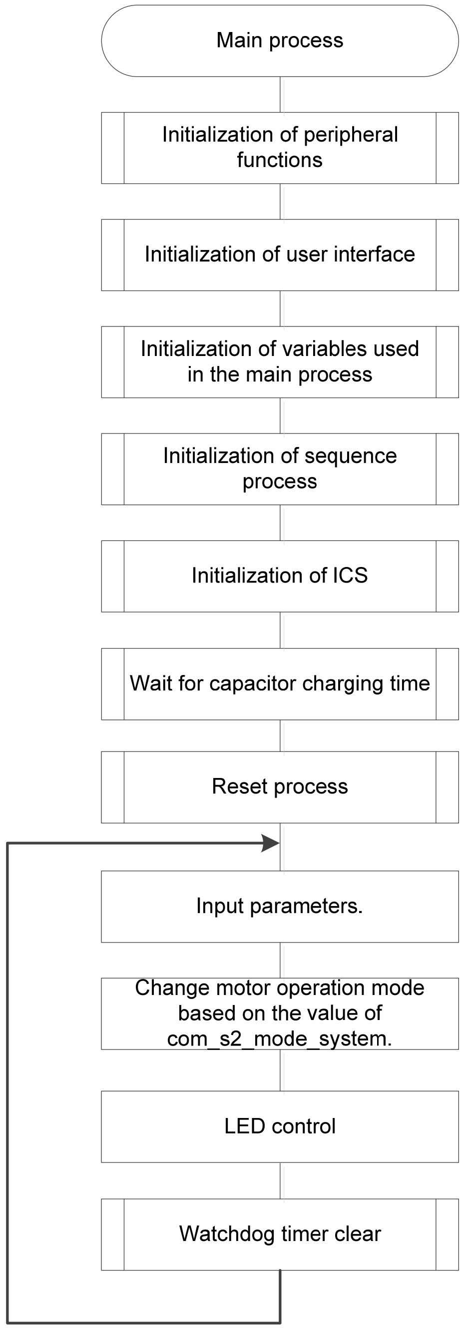

Figure 3-6 Main Process Flowchart (Target Software: (2))

3.6.2 100 [μs] cycle interrupt handling (Sensorless Vector Control)

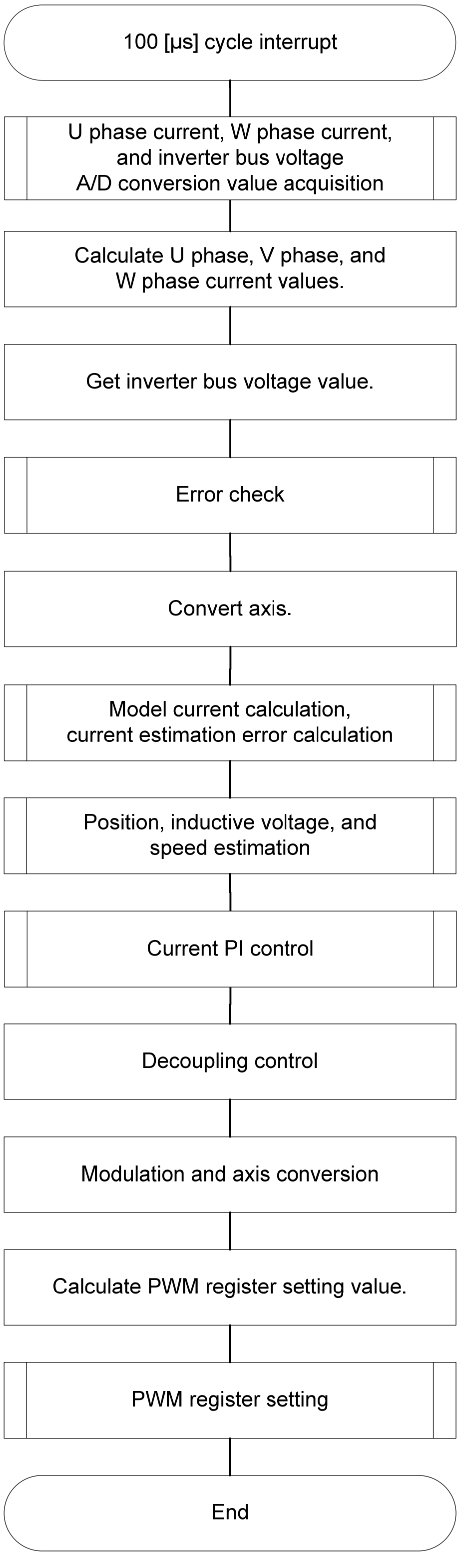

Figure 3-7 100 [μs] Cycle Interrupt Handling (Sensorless Vector Control)

3.6.3 1 [ms] interrupt handling

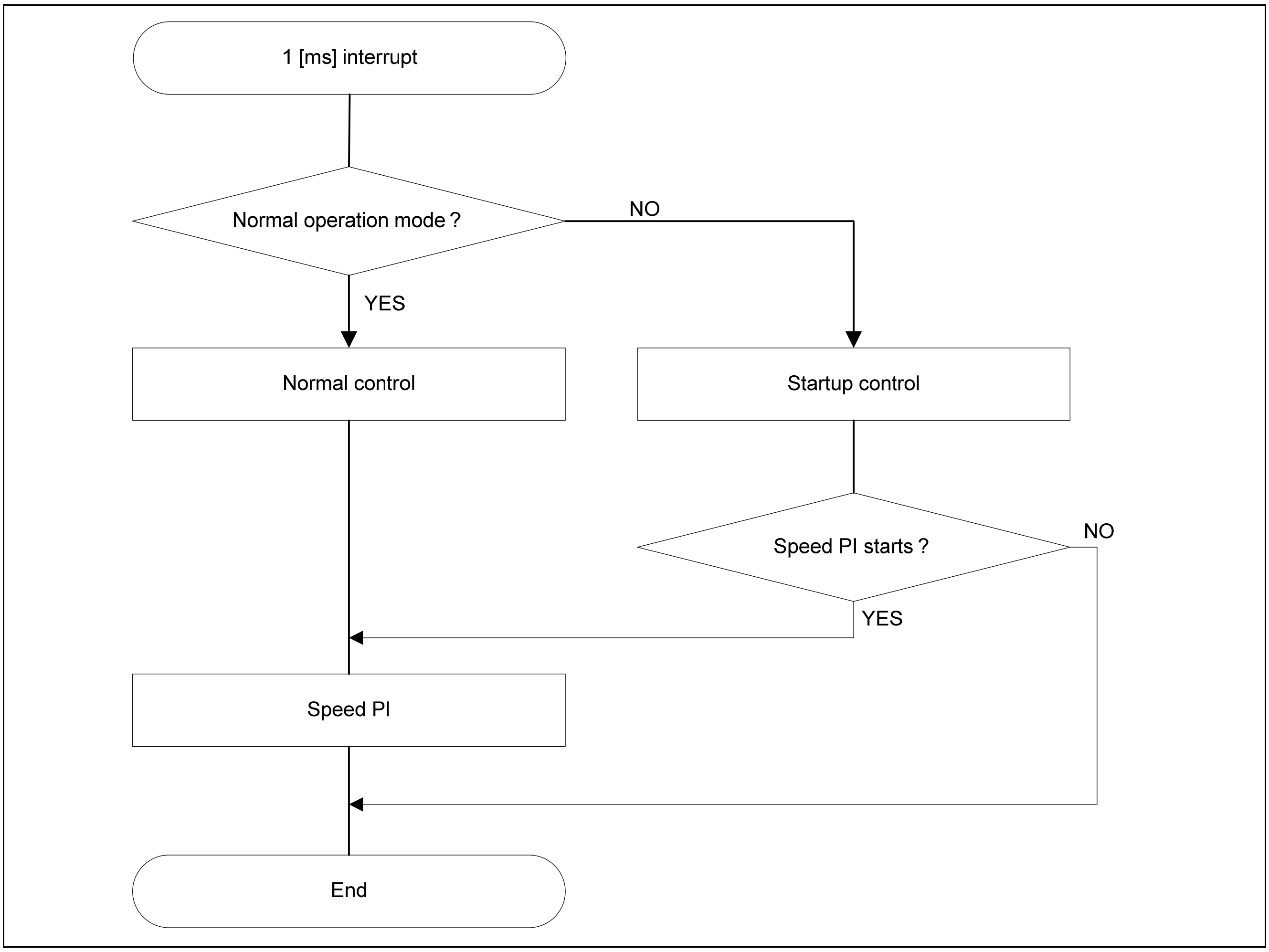

Figure 3-8 1 [ms] Interrupt Handling

3.6.4 Over current detection interrupt handling

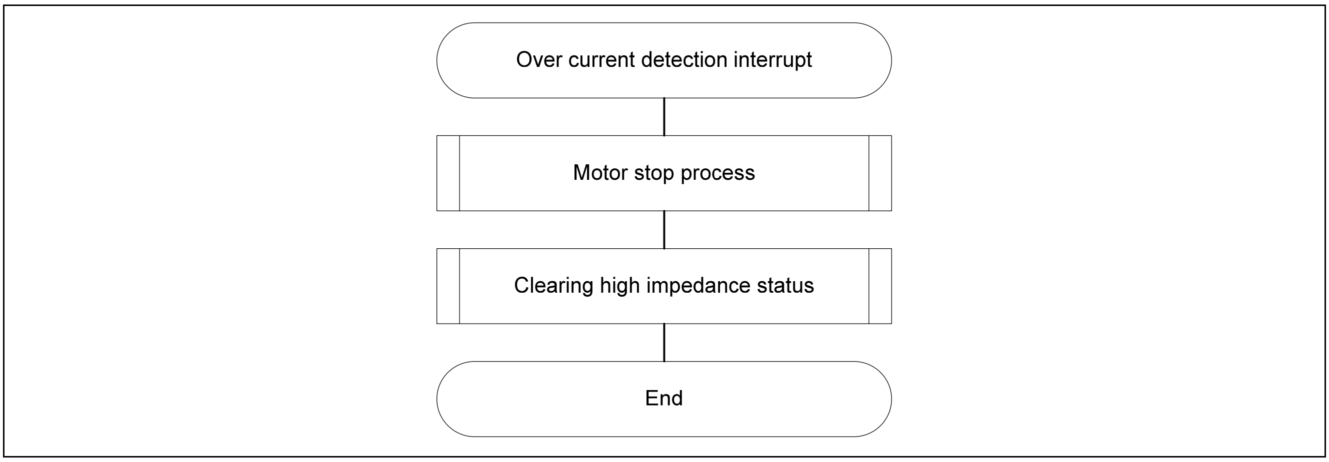

Figure 3-9 Over Current Detection Interrupt Handling

4. Development support tool In Circuit Scope

4.1 Overview

In the target sample programs described in this application note, user interfaces (start/stop command, rotation speed command, etc.) based on the development support tool 'In Circuit Scope' (ICS) can be used. ICS is a tool which displays real-time waveforms of global variables of the program being executed on the target system on PC. Refer to 'In Circuit Scope manual' for usage and more details.

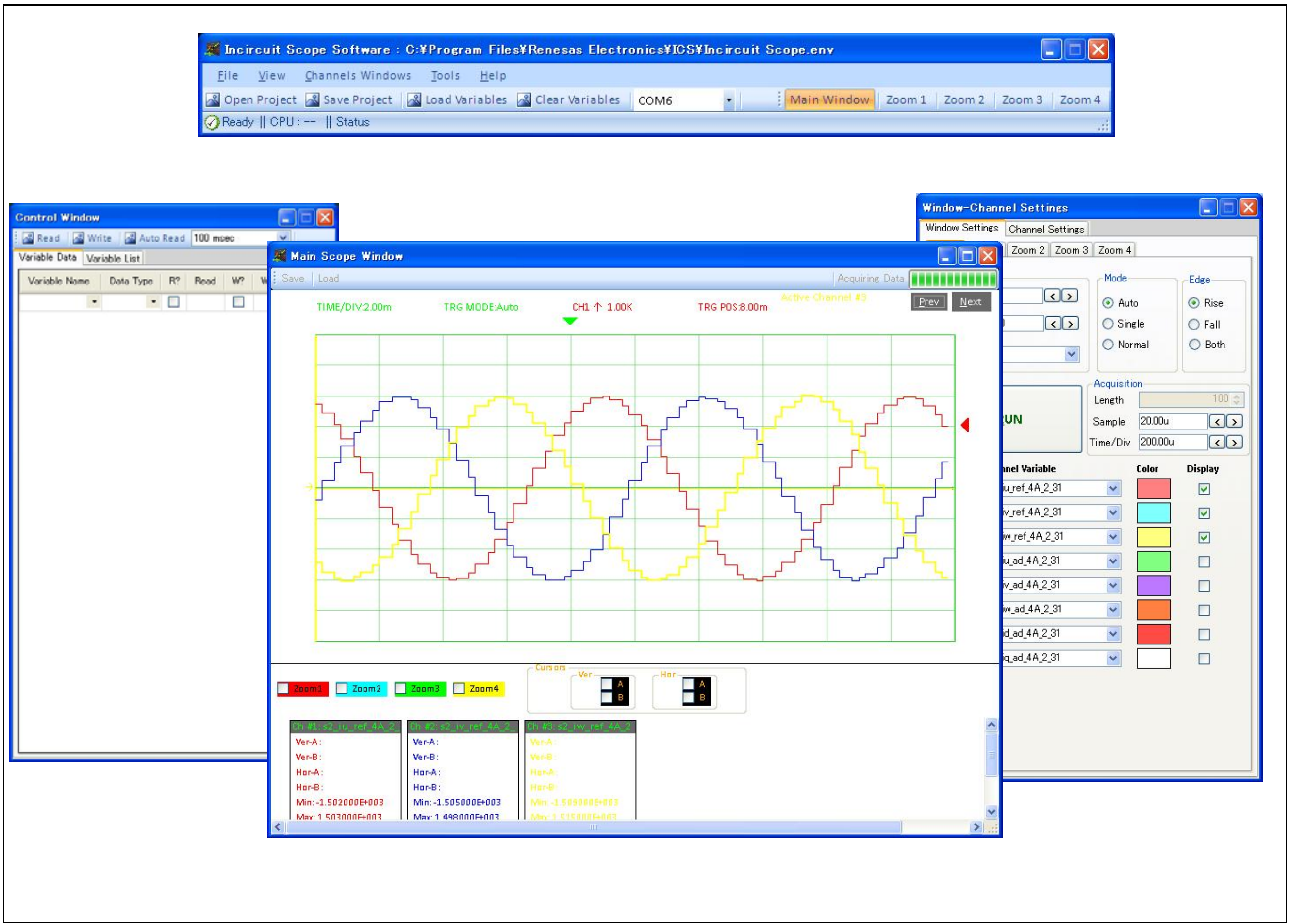

Figure 4-1 In Circuit Scope - Appearance

4.2 How to use the library

In order to use ICS for the low-voltage version, it is necessary to call functions related to ICS. The ICS-related functions have been set by conditional compilation (#ifdef--#endif). To use ICS, set as follows. As for the high-voltage version, ICS can be used without changing the source code.

[File name] mtr_common.h

[Point to change] Add the following declaration.

```
#define ICS_USE
```

4.3 List of variables for ICS

Table 4-1 is a list of variables for ICS. When these variables are changed, they are not reflected to variables in the motor control layer yet. However, the variables in the motor control layer are rewritten when the same values as g_s2_enable_write are written to com_s2_enable_write. Note that variables with (*) do not depend on com_s2_enable_write.

Table 4-1 List of Variables for ICS (1/1)

Variable name	Type	Content	Reflection destination variable (variable in motor control layer)
com_s2_sw_userif (*) (target software: (1))	int16	User interface switch 0: ICS user interface use (default) 1: Board user interface use	Reflected to g_s2_sw_userif upon rewriting
com_s2_mode_system(*)	int16	State management 0: Stop mode 1: Run mode 3: Reset	Reflected to g_s2_mode_system upon rewriting
com_s2_direction	int16	Rotation direction 0: CW 1: CCW	g_u1_direction
com_s2_ref_speed_rpm	int16	Speed command value (mechanical angle) [rpm]	g_s2_ref_speed
com_f4_kp_speed	float32	Speed PI control proportional term gain	g_f4_kp_speed
com_f4_ki_speed	float32	Speed PI control integral term gain	g_f4_ki_speed
com_f4_kp_id	float32	d axis current PI control proportional term gain	g_f4_kp_id
com_f4_ki_id	float32	d axis current PI control integral term gain	g_f4_ki_id
com_f4_kp_iq	float32	q axis current PI control proportional term gain	g_f4_kp_iq
com_f4_ki_iq	float32	q axis current PI control integral term gain	g_f4_ki_iq
com_f4_k_emf	float32	Speed electromotive force estimation gain	g_f4_k_emf
com_f4_k_theta	float32	Position estimation gain	g_f4_k_theta
com_f4_speed_lpf_k	float32	Speed LPF parameter	g_f4_speed_lpf_k
com_f4_current_lpf_k	float32	Current LPF parameter	g_f4_current_lpf_k
com_f4_mtr_r	float32	Resistance [Ω]	mtr_p
com_f4_mtr_l	float32	Inductance [H]	mtr_p
com_f4_mtr_m	float32	Flux [Wb]	mtr_p
com_f4_offset_lpf_k	float32	LPF parameter of current offset value	g_f4_offset_lpf_k
com_s2_max_speed_rpm	int16	Maximum speed value (mechanical angle) [rpm]	g_s2_max_speed
com_s2_ol_to_less_speed_rpm	int16	Sensorless control switching speed (mechanical angle) [rpm]	g_s2_ol_to_less_speed_rad
com_s2_ol_iq_up_speed_rpm	int16	Speed at start of q axis current command value increase (mechanical angle) [rpm]	g_s2_ol_iq_up_speed _rad
com_s2_less_to_ol_speed_rpm	int16	Open loop switching speed (mechanical angle) [rpm]	g_s2_less_to_ol_speed_rad
com_f4_ol_ref_id	float32	Command d axis current in open loop mode [A]	g_f4_ol_ref_id g_f4_ol_id_up_step g_f4_id_down_step
com_f4_ol_id_up_time	float32	Command d axis current adding time in open loop mode [ms]	
com_f4_id_down_time	float32	Command d axis current subtracting time [ms]	
com_f4_ref_speed_const_time	float32	Time during which speed command value is constant [ms]	g_f4_ref_speed_const_time
com_f4_accel	float32	Acceleration [rad/s^2]	g_f4_accel
com_f4_fluctuation_limit	float32	Speed fluctuation limit value [rad/s]	g_f4_fluctuation_limit
com_f4_ol_iq_down_time	float32	Command q axis current subtracting time [ms]	g_f4_ol_iq_down_step g_f4_ol_ref_iq g_f4_ol_iq_up_step
com_f4_ol_ref_iq	float32	q axis current command value in open loop [A]	
com_f4_ol_iq_up_time	float32	q axis current command value adding time in open loop mode [ms]	
com_f4_offset_calc_time	float32	Current offset value calculation time [ms]	g_f4_offset_calc_time
com_s2_enable_write	int16	Enabled to rewriting variables	

Website and Support

Renesas Electronics Website
http://www.renesas.com/

Inquiries
http://www.renesas.com/contact/

Revision Record

Rev.	Date of issue	Descriptions	
		Page	Summary
1.00	2014/09/26	—	First edition issued

General Precautions in the Handling of MPU/MCU Products

The following usage notes are applicable to all MPU/MCU products from Renesas. For detailed usage notes on the products covered by this document, refer to the relevant sections of the document as well as any technical updates that have been issued for the products.

1. Handling of Unused Pins
   Handle unused pins in accord with the directions given under Handling of Unused Pins in the manual.
   — The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible. Unused pins should be handled as described under Handling of Unused Pins in the manual.
2. Processing at Power-on
   The state of the product is undefined at the moment when power is supplied.
   — The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the moment when power is supplied.
   In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the moment when power is supplied until the reset process is completed.
   In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the moment when power is supplied until the power reaches the level at which resetting has been specified.
3. Prohibition of Access to Reserved Addresses
   Access to reserved addresses is prohibited.
   — The reserved addresses are provided for the possible future expansion of functions. Do not access these addresses; the correct operation of LSI is not guaranteed if they are accessed.
4. Clock Signals
   After applying a reset, only release the reset line after the operating clock signal has become stable. When switching the clock signal during program execution, wait until the target clock signal has stabilized.
   — When the clock signal is generated with an external resonator (or from an external oscillator) during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Moreover, when switching to a clock signal produced with an external resonator (or by an external oscillator) while program execution is in progress, wait until the target clock signal is stable.
5. Differences between Products
   Before changing from one product to another, i.e. to a product with a different type number, confirm that the change will not lead to problems.
   — The characteristics of an MPU or MCU in the same group but having a different part number may differ in terms of the internal memory capacity, layout pattern, and other factors, which can affect the ranges of electrical characteristics, such as characteristic values, operating margins, immunity to noise, and amount of radiated noise. When changing to a product with a different part number, implement a system-evaluation test for the given product.

SALES OFFICES Renesas Electronics Corporation http://www.renesas.com

Refer to "http://www.renesas.com/" for the latest and detailed information.

Renesas Electronics America Inc.
2801 Scott Boulevard Santa Clara, CA 95050-2549, U.S.A.
Tel: +1-408-588-6000, Fax: +1-408-588-6130

Renesas Electronics Canada Limited
1101 Nicholson Road, Newmarket, Ontario L3Y 9C3, Canada
Tel: +1-905-898-5441, Fax: +1-905-898-3220

Renesas Electronics Europe Limited
Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K
Tel: +44-1628-585-100, Fax: +44-1628-585-900

Renesas Electronics Europe GmbH
Arcadiastrasse 10, 40472 Düsseldorf, Germany
Tel: +49-211-6503-0, Fax: +49-211-6503-1327

Renesas Electronics (China) Co., Ltd.
Room 1709, Quantum Plaza, No.27 ZhiChunLu Haidian District, Beijing 100191, P.R.China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

Renesas Electronics (Shanghai) Co., Ltd.
Unit 301, Tower A, Central Towers, 555 Langao Road, Putuo District, Shanghai, P. R. China 200333
Tel: +86-21-2226-0888, Fax: +86-21-2226-0999

Renesas Electronics Hong Kong Limited
Unit 1601-1613, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2265-6688, Fax: +852 2886-9022/9044

Renesas Electronics Taiwan Co., Ltd.
13F, No. 363, Fu Shing North Road, Taipei 10543, Taiwan
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670

Renesas Electronics Singapore Pte. Ltd.
80 Bendemeer Road, Unit #06-02 Hyflux Innovation Centre, Singapore 339949
Tel: +65-6213-0200, Fax: +65-6213-0300

Renesas Electronics Malaysia Sdn.Bhd.
Unit 906, Block B, Menara Amcorp, Amcorp Trade Centre, No. 18, Jln Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia
Tel: +60-3-7955-9390, Fax: +60-3-7955-9510

Renesas Electronics Korea Co., Ltd.
12F., 234 Teheran-ro, Gangnam-Ku, Seoul, 135-920, Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5141

Colophon 4.0